



2019

חומריע עזר בראשותה תקשורת כיתה י"ג



תוכן

7	Erasing and reloading the Switch
10	מתק (switch)
12	DHCP Server
16	Port Security
19	Virtual LAN (VLAN)
20	פרישת VLAN על פני כמה Switches
22	חיבור בין VLANS
23	VLAN Trunking Protocol (VTP)
24	הגדרת VTP
25	Access Control Lists (ACL)
27	Standard ACL
28	Extended ACL
29	Named ACL
31	Static Routing
34	Routing Protocols
36	Routing Information Protocol (RIP)
36	הגדרת RIP
37	בדיקות ההגדרות
38	Subnetting IP Networks
41	Type of NAT Addresses
41	NAT Overload (PAT)
43	Static NAT
43	Port Forward
44	Dynamic NAT
45	Internet Protocol Version 6 (IPv6)
47	סוגי כתובות
47	(FF00::/8) Multicast
47	(one to nearest) Anycast
48	סוגי Unicast Addresses
48	(FE80::/10) Link-Local
48	(2000::/3) Unique Global
48	(FC00::/7) Unique local

48(::1) Loopback
49Extended Unique Identifier (EUI-64)
50Neighbor Discovery Protocol (NDP)
51DHCPv6
53מעבר כתובות מי IPv4 ל- IPv6
54IPv6 Subneting
55Cisco Discovery Protocol (CDP)
56Link Layer Discovery Protocol (LLDP)
57Network Time Protocol (NTP)
58System Message Logging (Syslog)
60Spanning Tree Protocol
68Open Shortest Path First (OSPF)
77Enhanced Interior Gateway Routing Protocol (EIGRP)
77 תכונות EIGRP
77 טבלאות EIGRP
77 טבלת השכנים (neighbors) שcn זהו נתב שתומך ב- EIGRP וגם מחובר לנtab שלנו באותו Broadcast Domain
78 טבלת הטופולוגיה
79 כיצד EIGRP מחשב את ה- Metric של נתיב?
80 בחירה הנתיב הטוב ביותר
80 EIGRP Packets
81 הגדרת EIGRP
82 Auto Summarization
82 Passive Interface
83 הגדרת authentication (הגברת רמת האבטחה)
84 Wide Area Networks (WAN)
85 WAN Topology
86 Physical Layer Terminology
87 WAN Technologies
90 (High-Level Data-Link Control) HDLC
92 Layer 3 Redundancy Protocol - HSRP
94 הגדרת HSRP
94 Interface Tracking
94 Preempt Option
94 שינוי Timers
95 HSRP Load Balancing
95 HSRP Troubleshooting

96	Simple Network Management Protocol (SNMP)
96	רכיבי SNMP
96	גרסאות SNMP
97	פקודות של SNMP
97	Monitoring applications
97	הזרת SNMP
99	Virtual Private Network
100	VPN Protocols
100	VPN Authentication Protocols
102	Internet Protocol Security (IPSec)

Basic IOS Configuration

Base Command Modes		
Switch>	מצב משתמש ללא זכויות	User Mode
Switch#	זכויות, בעיקר לצפייה בהגדירות	Privileged EXEC mode
Switch(config)#	אפשר שינוי הגדירות כלליות	Global configuration mode
Switch(config-if)#	אפשר שינוי הגדירות הממשק	Interface Configuration Mode

מעבר בין מצבים עובודה	
Switch>enable	כניסה ל- Enable Mode
Switch#configure terminal	כניסה ל- Global configuration mode
Switch(config)#interface {fastEthernet} {0/1}	כניסה ל- Interface configuration mode
Switch#exit	חזרה מכל מצב, רמה אחת אחרת
Switch(config-if)#end	.Enable Mode
CTRL+Z	כניסה ל- Enable Mode

במצב משתמש ניתן לבצע מגוון פקודות שונות. שים לב שאתה נימצא במצב משתמש המתאים לפניה שימוש בפקודה. קיימים מצבים שאינם מופיעים, אותם נלמד בהמשך.

פקודות בסיסיות	
Switch#show history	显示历史命令
Switch(config)#hostname {name}	设置主机名
Switch#show version	显示版本信息
Switch#show running-config	显示当前配置 (RAM)
Switch#show startup-config	显示启动配置 (NVRAM)

Backup and Restore

שמירת ההגדירות שרצות כרגע (ההגדירות ב- RAM) ל- :NVRAM (RAM) ל-

(כדי שההגדירות יעלו כל פעם עם אתחול המתג)

Switch#copy running-config startup-config

Destination filename [startup-config]?[Enter]

העתיקת ההגדירות מ- RAM ל- NVRAM

Switch#copy startup-config running-config

Destination filename [running-config]?[Enter]

Banner

Banner זו הודעה שתופיע בעת העלאת המתג/הנתב. ההודעה חייבת להתחיל בתו מסוים (לדוגמא @) ולהסתיים באותו תו.

Switch(config)#banner motd @ {ההודעה} @

דוגמא:

Switch(config)#banner motd @

#####
The Administrator is YAKI
#####
Switch>

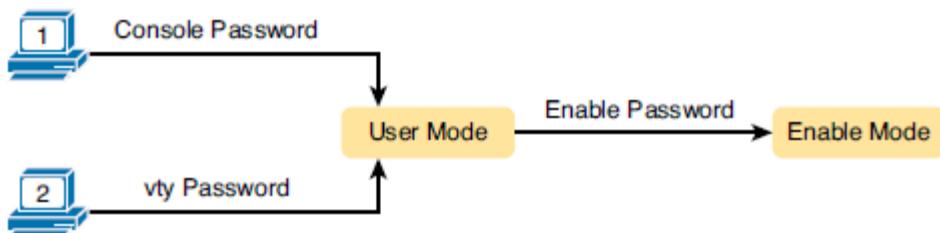
#####
The Administrator is YAKI
#####
Switch>

@

אבטחת הכניסה למתק/נתב

הגדרת סיסמאות

בכדי לאבטח את הכניסה למתק יש צורך להגדיר סיסמאות כניסה.



סיסמת כניסה ל- (User Mode) Console :

```
Switch(config)#line console 0
Switch(config-line)#password {password}
Switch(config-line)#login
```

סיסמת כניסה ל- (User Mode) Virtual Terminal Access
ניתן לפתח עד 16 חיבורים של Telnet במקביל.

```
Switch(config)#line vty 0 4
Switch(config-line)#password {password}
Switch(config-line)#login
```

סיסמת כניסה ל- Enable Mode :

```
Switch(config)#enable password {password}
```

הצפנה והסתרת הסיסמאות

הסיסמאות לא מוצפנות, ניתן לראות אותן באמצעות הפקודה config

הצפנה כל הסיסמאות שניתנו/יינטנו (ההצנה חלה וניתנת לפריצה)
Switch(config)#service password-encryption

סיסמת כניסה מוצפנת ל- Enable Mode :

הסיסמה חתומה דיגיטלית באמצעות אלגוריתם Message Digest 5 (MD5).
כלומר במקום הסיסמה, מואחסן hash של הסיסמה. enable secret יותר מ-
enable password, כלומר זו הסיסמה שהמשתמש יצרך להכנסו.
Switch(config)#enable secret {password}

ניהול מtag/נתב מרוחק

הגדרת תמיכה ב- Telnet

כדי לנוהל מtag מרוחק, יש צורך לחת כנתובת לחת ל- `vlan`, דרכו נתחבר למtag.
כברירת מחדל כל הממשקים שייכים ל 1 `vlan` אך מטעמי אבטחה, עדיף לנוהל את המtag
דרך `vlan` אחרת מי 1 `vlan`. בנוסף יש לחת סימא ל- `vty`.

```
Switch(config)#interface vlan {1}
Switch(config-if)#ip address {ip} {subnet mask}
Switch(config-if)#no shutdown
```

כדי לאפשר התחברות למtag מרשט שונה, נגדיר למtag :

```
Switch(config)#ip default-gateway {ip}
```

דוגמא, התחברות למtag ממtag אחר:

```
SW01#telnet 10.0.0.200
Trying 10.0.0.200 ...Open
```

```
User Access Verification
```

```
Password:
SW02>
```

כדי לסיים את ההתקשרות ב- Telnet, נרשם `.exit`.
ניתן לחזור ל- console ללא יציאה מההתקשרות, נלחץ **Ctrl+Shift+6** ולאחר מכן נלחץ **X**.

כדי לראות איזה התקשרות Telnet פתוחה, נרשם `show sessions`
* מסמנת את ההתקשרות الأخيرة, לחיצה על `Enter` תחזיר אותה לשם.
לחיצה על מספר ה- `connection` Enter ו- `connection` אותו ל- (לא נתמך ב-
(packet tracer)

```
SW01>show sessions
Conn Host Address Byte Idle Conn Name
  1 10.0.0.200 10.0.0.200 0 0 10.0.0.200
* 2 10.0.0.150 10.0.0.150 0 0 10.0.0.150
```

כדי לראות מי מחובר למtag, נרשם `show users`

```
SW02>show users
Line User Host(s) Idle Location
* 0 con 0 idle 00:00:00
  2 vty 0 idle 00:05:49 10.0.0.100
```

```
Interface User Mode Idle Peer Address
```

כדי לנתק חיבור, נרשם {2} `disconnect`

הגדרת תמיכה ב- SSH

SSH מאפשר ניהול מרוחק בצורה מוצפנת, בניגוד ל- Telnet (לא מוצפן).

Switch01(config)# username {yaki} password {1234}	নির্দেশনা
Switch01(config)# ip domain-name {yaki.local}	Domain Name
Switch01(config)# crypto key generate rsa	יצירת המפתחות (רצוי באורך של 1024)
Switch01(config)# ip ssh version 2	קביעת גרסה SSH שאתה אנו רוצים לעבוד
Switch01(config)# line vty 0 4	שינוי תמיכה מי Telnet ל- SSH
Switch01(config-line)# transport input {ssh/telnet/all/none}	

פקודת הכניסה שנרשום במחשב כדי להיכנס למוג.

ssh -l {user name} {IP}

הסיסמה שהמשתמש מתבקש להכניס היא של ה- VTY.

Erasing and reloading the Switch

1. אם קיימ קובץ הגדרות vlan, יש למחוק אותו:

```
Switch#delete flash:vlan.dat  
Delete filename [vlan.dat]?[enter]  
Delete flash:vlan.dat? [confirm] [enter]
```

אם לא שומר קובץ vlan, תתקבל הודעה הבאה:

```
%Error deleting flash:vlan.dat (No such file or directory)  
2. מחיקת קובץ ההגדרות השמור ב- NVRAM
```

```
Switch#erase startup-config
```

```
Erasing the nvram filesystem will remove all files! Continue? [confirm]  
Erase of nvram: complete
```

אם לא שמרו הגדרות ב- nvram, תתקבל הודעה הבאה:

```
%% non-volatile configuration memory is not present
```

3. אתחול מחדש של ה- switch

```
Switch#reload  
System configuration has been modified. Save? [yes/no]:N  
Proceed with reload? [confirm] [Enter]  
Reload requested by console.  
Would you like to enter the initial configuration dialog? [yes/no]:N  
Press RETURN to get started! [Enter]
```

Recover access to the switch

1. כבה את ה- switch. הדלק אותו תוך כדי להיצה רצופה על הכפטור "MODE" שנמצא בפנل הקדמי של ה- switch.
2. יש להקליד את הפקודות הבאות:

Switch:flash_init

Switch:load_helper

Switch:dir flash:

3. שינוי שם קובץ ההגדרות לשם זמני (הקובץ מכיל כל ההגדרות כולל הסיסמות):

Switch#rename flash:config.text flash:config.old

4. בצע שוב dir כדי לבדוק שם הקובץ שונה מי config.text ל config.old

Switch:dir flash:

5. אתחול מחדש של המערכת:

Switch:boot

Continue with the configuration dialog? [yes/no] : **N**

6. שינוי שם קובץ ההגדרות לשמו המקורי:

Switch#rename flash:config.old flash:config.text

7. העתקת קובץ ההגדרות לזכרון:

Switch#copy flash:config.text system:running-config

Source filename [config.text]?[enter]

Destination filename [running-config][enter]

8. טעינת קובץ ההגדרות והשלמה. הפקודות הבאות מבטלות את הסיסמות הישנות:

Switch(config)#no enable secret

Switch(config)#no enable password

Switch(config)#line console 0

Switch(config-line)#no password

Switch(config-line)#no login

Switch(config-line)#exit

Switch(config)#line vty 0 15

Switch(config-line)# no password

Switch(config-line)#no login

הגדרת ממשק fastEthernet

כדי לראות את מצב הממשקים:

Router#show ip interface brief

הגדרת ממשק fastEthernet בנתב

Router(config)#interface {fastEthernet} {0/0}

Router(config-if)#ip address {172.18.0.254} {255.255.0.0}

Router(config-if)#no shutdown

הגדרת מהירות הממשק:

```
Switch(config-if)#speed {auto/10/100/1000}
```

הגדרת duplex

```
Switch(config-if)#duplex {auto/full/half}
```

הגדרה בתקנון שמות באנטזיה DNS

להגדרת כתובת שרת DNS, משתמש בפקודה:

```
Switch(config)#ip name-server {dns address}
```

הגדרת שימוש בשרת DNS (מודדר כברירת מחדל):

```
Switch(config)#ip domain-lookup
```

כאשר מתג/נתב מנסה לתרגם שם ולא מצליח, המערכת נתקעת למשך 30 שניות.

כדי למנוע מהתקן לנסוט לתרגם שמות משתמש להשתמש בפקודה:

```
Switch(config)#no ip domain-lookup
```

סנכרון ההודעות

כשנשלחת הודעה למסך, לעיתים היא מפיער בכתיבת פקודה.

הפקודה מסנכרנת הודעות שהנתב/מתג שלוח למסך, כך שלא יפריעו לכתיבה פקודה.

```
Switch(config)#line console 0
```

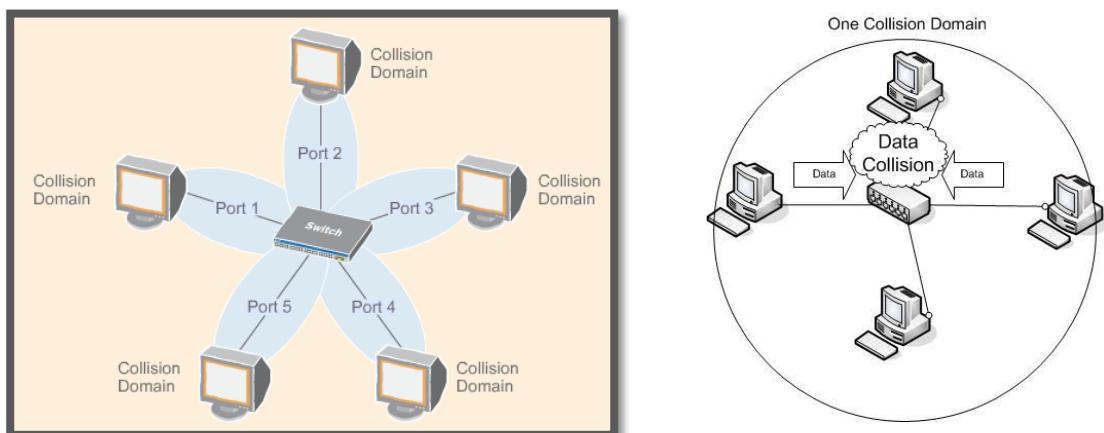
```
Switch(config-line)#logging synchronous
```

הערה:

ניתן לרשום את ההגדרות בכתבן ולאחר מכן להכניס לנתב/מתג.

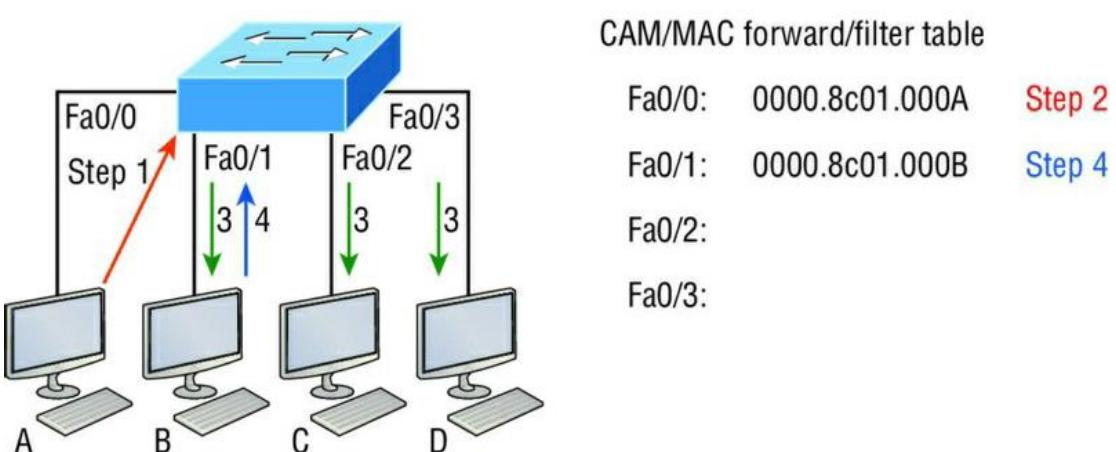
מtag (switch)

מtag פועל בשכבה השנייה של מודול OSI והוא מזהה את התקנים שמחוברים אליו לפי כתובות פיסיות. כל מסוק של המtag יוצר מרחב התנגשות (collision domain) נפרד. Collision Domain מגדר אזור ברשות שבו אסור לשני מחשבים לשדר בו זמני. ככל שמתחם ההתנגשות גדול יותר כך גדל הסיכוי להתנגשות בין שתי חבילות מידע. התנגשות גורמת להפסקת התקשורת לפרק זמן אקריאי (בסדר גודל של אלףות שנייה) בכל מתחם ההתנגשות. אחת הדרכים לשפר את ביצוע הרשת, היא לחלק את הרשת לכמה שיטות Collision Domain ובכך להקטין את הסיכוי לצירוף ההתנגשויות. מתחם ההתנגשות נת含ם על ידי מטגים ונתבים.



המtag מתחזק טבלת כתובות שנקראת (CAM) Content Addressable Memory שמכלילה כתובות של התקנים שמחוברים אליו. כאשר המtag נדלק הטבלה ריקה. כאשר חבילה (packet) נכנסת למtag, המtag קורא את כתובות המקור ומעדכן כך את הטבלה. אם המtag מקבל מנה שכתובת היעד שללה לא מופיע בטבלת הכתובות, המtag שלח את המנה דרך כל היציאות שלו (flooding) מלבד הממשק דרכו נכנסה החבילה.

1. מחשב A שולח חבילה למחשב B.
2. המtag לומד את הכתובת של מחשב A ומעדכן זאת בטבלה.
3. המtag לא מזהה את כתובת היעד של מחשב B בטבלה ולכן המtag שולח את החבילה דרך כל היציאות שלו מלבד הממשק ממנו החבילה נכנסה (Flooding).
4. מחשב B מגיב על ידי שליחת חבילה חוזרת למחשב A.
5. המtag לומד את הכתובת של מחשב B ומעדכן זאת בטבלה.



כדי לראות את טבלת הכתובות נשתמש בפקודה: **show mac address-table**

```
Switch#show mac address-table
      Mac Address Table
-----
Vlan      Mac Address          Type      Ports
----      -----
 1        0001.c743.cae3    DYNAMIC   Fa0/2
 1        0001.c922.b898    DYNAMIC   Fa0/3
 1        0002.17d7.bbd5    DYNAMIC   Fa0/1
 1        0005.5e6d.e1d8    DYNAMIC   Fa0/4
```

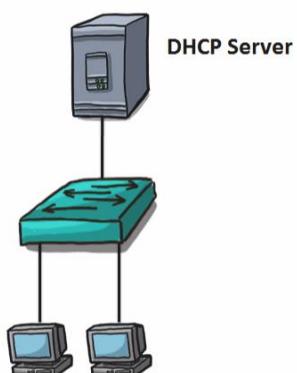
DHCP Server

Dynamic Host Configuration Protocol

Private IP addresses

כתובות פרטיות מיועדות לשימוש בתחום הרשת המקומי ולא ניתן לגשת אותן לאינטרנט. כתובות הרשת שמודגרות כתובות פרטיות הן:

- 10.0.0.0/8 - Class A
- 172.16.0.0/16 - 172.31.0.0/16 - Class B
- 192.168.0.0/24 - 192.168.255.0/24 - Class C

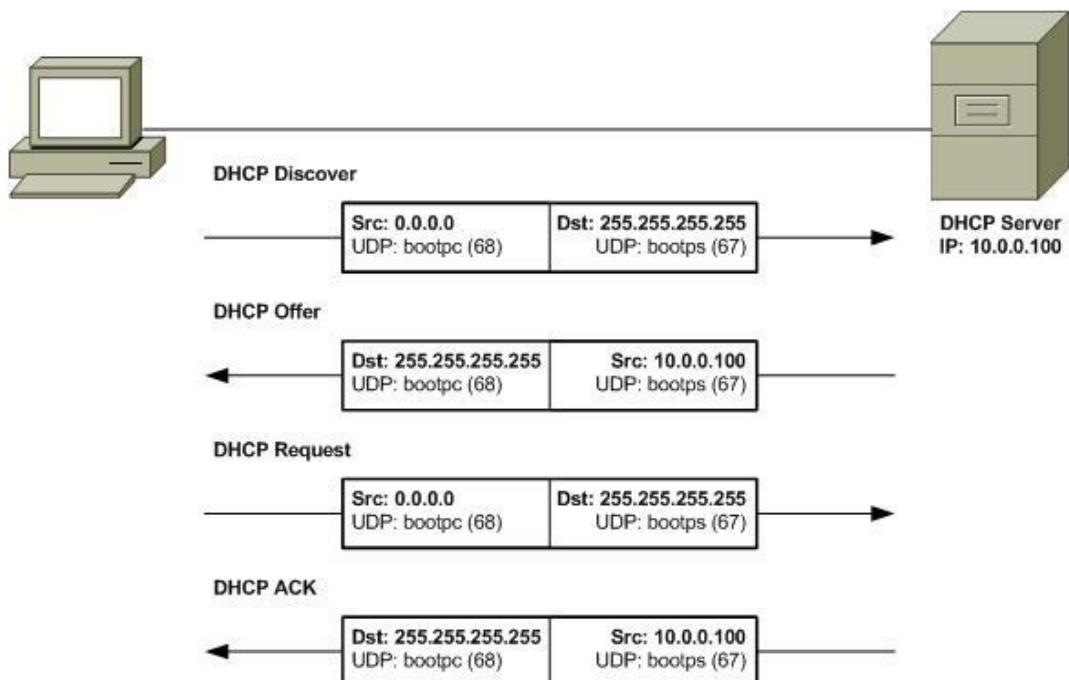


שרת DHCP מחלק כתובות פרטיות להתקנים ברשת LAN.

תהליך קבלת כתובת IP משרת DHCP
התהליך כולל ארבע שלבים:

1. DHCP discover - הלקוח מחפש שרת DHCP.
2. DHCP offer - השרת מציאה כתובת ללקוח.
3. DHCP request - הלקוח מאשר את ההצעה.
4. DHCP acknowledgement - השרת מאשר את הבקשה.

כל התהליך מתבצע ב-.broadcasts. השרת יציג על-ידי udp port 67 והלקוח מיוצג על-ידי udp port 68.



חידה: איזה שלב מיצג ה- packet הבא?

```
Ethernet II, Src: HonHaiPr_da:cb:a3 (48:5a:b6:da:cb:a3), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 68 (68), Dst Port: 67 (67)
Bootstrap Protocol (Discover)
```

לקוח	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover
שרות	192.168.1.1	255.255.255.255	DHCP	342	DHCP Offer
לקוח	0.0.0.0	255.255.255.255	DHCP	351	DHCP Request
שרות	192.168.1.1	255.255.255.255	DHCP	342	DHCP ACK

מה קורה אם השירות DHCP לא זמין או שלא נשארו לו כתובות IP לחלק?
 אם לאחר שליחת ארבע Discover packets לא התקבלה תשובה מהשירות, הלוקוח יתנו
 לעצמו כתובת מסוג (APIPA) (Automatic Private IP Addressing).
 זו כתובת קו מתוך כתובות רשת 169.254.0.0/16.
 כדי לנסוט לקלט כתובת קו, הלוקוח יסדר כל שלוש דקודות.

```
Autoconfiguration IP Address. . . : 169.254.79.112
Subnet Mask . . . . . : 255.255.0.0
Default Gateway . . . . . :
```

מה קורה כאשר יש יותר משלת DHCP אחד?
 כאשר הלוקוח משדר משלר packets Discover, כל התקן ברשות מקבל זאת ואם יש ברשות יותר
 משלת DHCP אחד הלוקוח קיבל כמה DHCP offer.
 הלוקוח יודע איזה הצעה מיועדת אליו כי בתוקף- Packet מופיעה הכתובת הפיסית של
 הלוקוח.
 כדי לאשר את ההצעה, הלוקוח שולח בקשה לשרת הראשון ממנו הוא קיבל offer packet.
 השירות שולח לлокוח אישור על כך שהכתובת שמורה לлокוח לתקופה זמן מסוימת (Lease).

הפקודה **ipconfig/all** מאפשרת לבדוק מה הכתובת של השירות DHCP ממנה התקבלה
 הכתובת ומה תקופת ה- Lease.

```
Lease Obtained. . . . . : Saturday, April 05, 2008 8:04:14 AM
Lease Expires . . . . . : Tuesday, January 19, 2038 6:14:07 AM
```

מה קורה כאשר נגמרה תקופת הזמן שהכתובת שומרה לлокוח ?(Lease)
 לאחר 50% מתקופת הזמן של ה- Lease, הלוקוח שולח לשרת request packet כדי להציג
 את ה- Lease. אם הלוקוח לא הצליח לתקשר עם השירות, אז לאחר 87.5% מתקופת הזמן,
 הלוקוח שולח Discover packet כדי לבקש כתובת קו משלת DHCP אחר.
 אם לאחר 100% מתקופה, אין שירות DHCP זמין, הלוקוח משתמש בכתובת מסוג A APIPA.

שליחת Release packet לשרת כדי לשחרר את הכתובת.
 כך השירות יכול להזכיר את הכתובת לлокוח אחר.

שליחת Discover packet בכך לחשוף שירות DHCP זמין – ipconfig/renew.

הגדרת הנטב לשירות DHCP

הגדרת Exclude

תחילה נגדיר איזה כתובות דן לא יחולקו. ניתן להגדיר Exclude גם לכתובות בודדת. רצוי לבצע Exclude לפני הגדרת הטווח כתובות לחלוקת זהאת כדי שהכתובות שאנו לא רוצים שיחולקו, לא יחולקו.

```
R1(config)#ip dhcp excluded-address {low IP} {high IP}
```

יצירת DHCP pool

כל הגדרות מתבצעות בתחום DHCP pool (טווח כתובות). ניתן להגדיר בנטב יותר מטוח כתובות אחד.

```
R1(config)#ip dhcp pool {pool name}
```

```
R1(dhcp-config)#

```

הגדרת כתובותחלוקת

הגדרת כתובת הרשות ממנה יחולקו הכתובות.

```
R1(dhcp-config)#network {network-id} {subnet-mask}
```

הגדרת כתובת נתב ברירה מחדל

```
R1(dhcp-config)#default-router {ip address}
```

הגדרת כתובת שירות DNS

ניתן להגדיר עד שמונה שרתי DNS כאשר הראשון הוא הראשי.

```
R1(dhcp-config)#dns-server {ip address}
```

הגדרת Lease (הפקודה לא נתמכת על-ידי Packet Tracer).

כברירת מחדל Lease מוגדר כיום אחד.

```
R1(dhcp-config)#lease {days} {hours} {minutes}
```

הגדרת domain name (הפקודה לא נתמכת על-ידי Packet Tracer).

```
R1(dhcp-config)#domain-name {domain name}
```

בדיקה, איזה כתובות חולקו:

```
Router#show ip dhcp binding
```

Conflict detection

יתכן שחלק מהכתובות ב- 100ק נמצאות בשימוש כתובות סטטיות. כדי לגלוות זאת ולא לחלק את הכתובות, השירות מנסה לבצע ping לפני חילוקה כתובות.

```
Router#show ip dhcp conflict
```

```
Router#clear ip dhcp conflict
```

DHCP Relay

נתבים לא מעבירים broadcasts ולכן לא יכול לקבל כתובת IP אם הוא נמצא ברשת שונה מזו של השרת. הגדרת הנטב כ-DHCP Relay מאפשרת לו לטווח בין הלקוח לשרת.

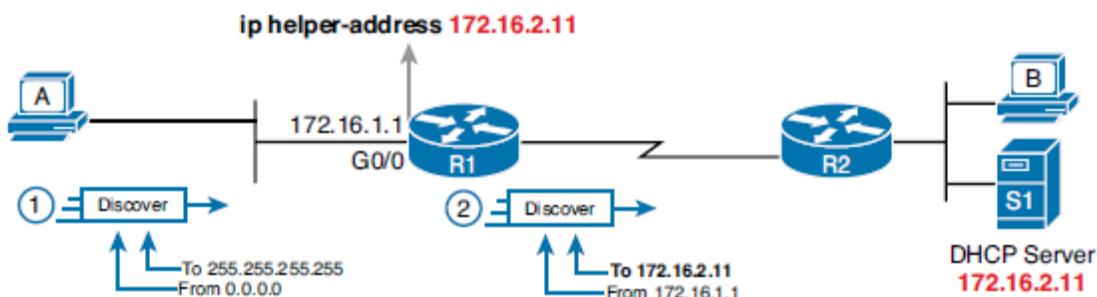
תהליך עבודת הנטב שמתפרק כ-:

1. ממתיין לבקשת מסווג כDiscover packets.
2. משנה את כתובת המקור של החבילה לכתובת של הממשק בנטב.
3. משנה את כתובת יעד לכתובת של השרת dhcp.
4. שולח את החבילה לשרת.

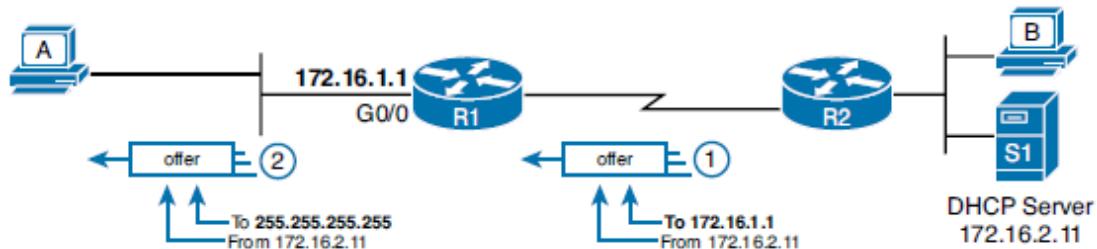
כדי להפוך את הנטב ל-:

Router(config-if)#ip helper-address {dhcp server ip address}

שליחת על-ידי הנטב Discover



שליחת על-ידי השרת offer



Port Security

זו טכנולוגיה שמאפשרת בקרה על התקנים שמתחברים לרשת. מtag לא יחבר לרשת התקן שלא מורשה להתחבר למשק. הטכנולוגיה מבוססת על סינון גישה על בסיס כתובות פיסיות. הטכנולוגיה פריצה על-ידי זיווף הכתובת הפיסית, אך לא לשכוו שהיא רק חלק ממכלול של הגנות ונחשבת כקו ההגנה הראשון מפני ניסיון חיבור התקנים לא מורשים לרשת.

מאפשר הגדרת שני מגבלות לכל משק:

- הגבלת מספר הרהתקנים שיוכלו להתחבר למשק במתג.
- קביעת הכתובת הפיסית שהמשק יŁmd.

מה זה violation?

כאשר התקן לא מורשה מתחבר למתג, מתרחשת הפרה (violation) של המדיניות. הפרה מתרחשת בשני מקרים:

- מחשב שלא מופיע במשק, מנסה לשדר דרך המשק.
- מחשב שנלמד במשק אחד, מנסה לשדר דרך ממשק אחר.

תגובה שמתרכשות בהפרה:

- **Shutdown** – המשק נסגר (הגדרת ברירת מחדל), מתבצע Log ונסלח SNMP .trap
- **Restrict** – המשק לא נסגר אך התקן זר לא מצליח להתחבר. מתבצע Log SNMP trap ונסלח .SNMP trap
- **Protect** – כמו Restrict אך לא Log ולא SNMP trap

Switch(config-if)# **switchport port-security violation** {shutdown/restrict/protect}

פתיחה PORT ב- violation מסווג

Switch(config-if)#**shutdown**
Switch(config-if)#**no shutdown**

Auto Recovery (Packet Tracer על-ידי)

אם התגובה להפרה במשק מוגדרת להיות Shutdown, אז ניתן להגדיר זמן בשניות להתחששות אוטומטית ממצב violation.

Switch(config)# **errdisable recovery cause psecure-violation**
Switch(config)# **errdisable recovery interval** {זמן בשניות}

הפעלה Port Security:

- כניסה למשק.
- הגדרת המשק כפועל במצב mode access .access mode dynamic access port - cabriraת מחדל המשק מוגדר ב- .port security מצב זה לא מאפשר פעולה Port Security .

Switch(config-if)#**switchport mode access**

- הפעיל את Port Security .

Switch(config-if)#**switchport port-security**

Maximum MAC addresses

הגדרת זו מגבילה את כמות הכתובות שמשק מוקן ללמידה (כברית מחדר כ-1)
Switch(config-if)# switchport port-security maximum {מספר}

MAC Address Learning

ניתן ללמד מתג בשני דרכים שונות, מי הム הכתובות ששייכות למשק.

Static Port Security

בשיטת זו, יש להכניס את הכתובת בצורה ידנית.

Switch(config-if)#switchport port-security mac-address {כתובת פיזית}

Sticky Port Security

בשיטת זו, הכתובת נלמדת בצורה דינמית.

Switch(config-if)#switchport port-security mac-address sticky

מחיקת כל הכתובות שנלמדו בשיטת sticky:

Switch#clear port-security sticky

מחיקת כתובות של משק מסוים:

Switch(config-if)#no switchport port-security mac-address sticky {mac_address}

כדי שהגדירות והכתובות ישמרו גם לאחר הפעלה מחדש של המתג, יש לשמר את הגדירות בעזרת הפקודה copy running-config startup-config.

MAC Address Aging

הגדירה, תוך כמה זמן ה- Port ישח את הכתובות שהוא למד ויחליף אותן בחדשות. שנות שני שיטות עובדה:

Absolute

זהרי הגדרת ברירת המחדל. כברית מחדר הזמן הוא 0, כלומר הכתובות לא ימחקו.

Inactivity

לאחר כמה דקות של חוסר פעילות ה- Port ישח את הכתובות.

כליום תקופת זמן בה המתג לא קיבל Frame מי ה- Port.

Switch(config-if)#switchport port-security aging type {inactivity/absolute}

Switch(config-if)#switchport port-security aging time {זמן בדקות}

Show

כדי לראות איזה כתובת למד כל port:

Switch#show port-security address

Secure Mac Address Table					
Vlan	Mac Address	Type	Ports	Remaining Age (mins)	
---	---	---	---	---	---
1	00E0.F7A6.13C5	DynamicConfigured	FastEthernet0/1	-	
1	00D0.BC64.0D77	DynamicConfigured	FastEthernet0/2	-	
1	0030.A389.C663	DynamicConfigured	FastEthernet0/3	-	

Total Addresses in System (excluding one mac per port) : 0
Max Addresses limit in System (excluding one mac per port) : 1024

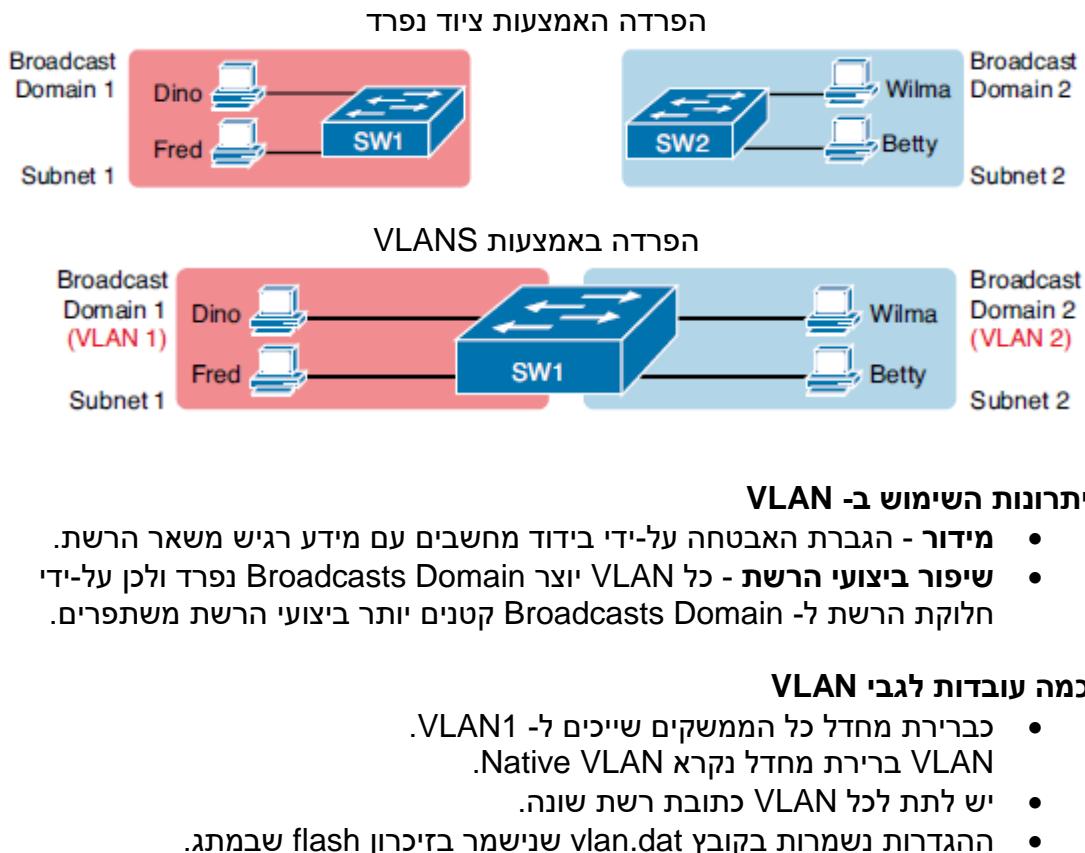
כדי לראות את הגדרות האבטחה של Port מסוים:

Switch#show port-security interface fastEthernet 0/1

Port Security	: Enabled
Port Status	: Secure-up
Violation Mode	: Shutdown
Aging Time	: 0 mins
Aging Type	: Absolute
SecureStatic Address Aging	: Disabled
Maximum MAC Addresses	: 1
Total MAC Addresses	: 1
Configured MAC Addresses	: 0
Sticky MAC Addresses	: 0
Last Source Address:Vlan	: 00E0.F7A6.13C5:1
Security Violation Count	: 0

Virtual LAN (VLAN)

הטכנולוגיה מאפשרת לחלק את הרשת לרשותות וירטואליות נפרדות. בתרשים ניתן לראות כיצד ניתן להשתמש במstag אחד לצורר הפרדה במקום לשני מטגים.



הגדרת VLAN

צירוף VLAN

```
Switch(config)#vlan {vlan-id}
Switch(config-vlan)#name {vlan-name} (לא חובה)
```

שיוך ממשק ל VLAN

```
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan {מספר}
```

שייך מספר ממשקים ל VLAN

```
Switch(config)#interface range fastethernet 0/{מספר}-{מספר}
Switch(config-if-range)#switchport access vlan {מספר}
```

הציגת מידע על VLANs

Switch#show vlan

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/2, Fa0/4, Fa0/5 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24
2 Sales	active	Fa0/3, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14

הציגת מידע על VLAN מסוימת:

Switch#show vlan name {vlan-name}

או

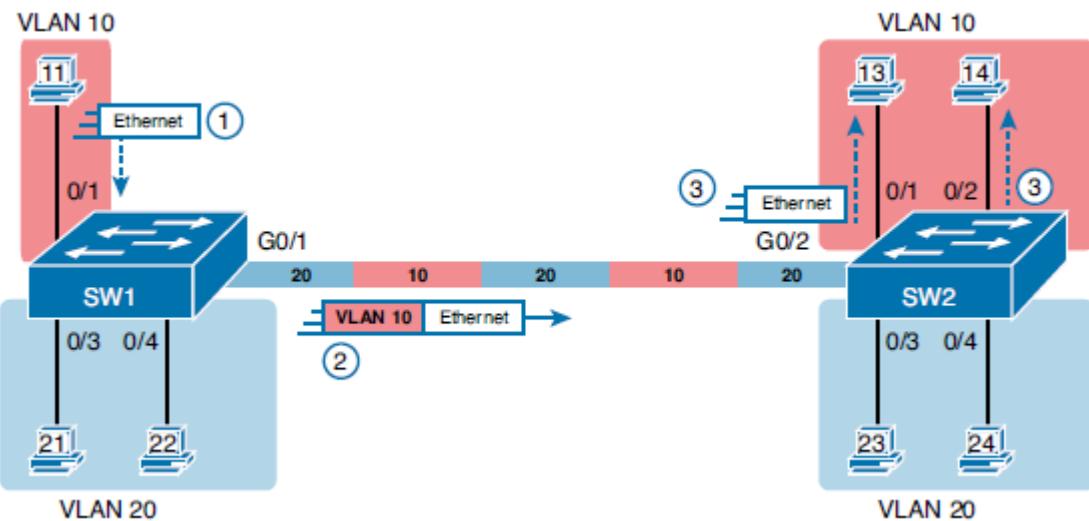
Switch#show vlan id {מספר}

Switch#show vlan id 2		
VLAN Name	Status	Ports
2 Sales	active	Fa0/3, Fa0/6, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14

פרישת VLAN על פני כמה Switches

ברשת שבה VLAN נפרש על פני כמה מתקנים, יש צורך בסימון כל Frame שעובר בין המתקנים, כדי שמתג היעד יידע לאיזה VLAN הוא Frame שייר. כדי לבצע זאת אנו מגדירים את ה- ports שמחברים בין המתקנים כ-Trunk.

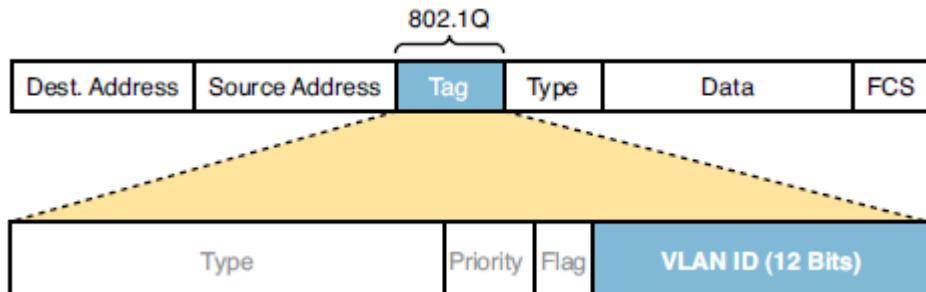
מסמן Frames Trunk Port על ידי הוספת TAG לכל Frame ובו מידע לאיזה VLAN שייר-he.



סוגי Trunking Protocols

קיימים שני פרוטוקולים שמבצעים Trunk:

- (ISL) Inter-Switch Link – נוצר על ידי Cisco הרבה לפני שונזר 802.1Q.
- כיוון הפרטוקול כמעט לא בשימוש ולא ניתן בחלק מהמתגים החדש של סיסקו.
- זה הפרטוקול שמשמש כבסיס מחדל במתגים. IEEE 802.1Q (dot1q)



Native VLAN

כך נקרא ה-VLAN שמשמש כבסיס מחדל. בברירת מחדל זהו VLAN1. Frame Trunking Protocol לא מוסיף Tag ל-Native VLAN Frame שמגיע מי Native VLAN וילך בשני המתגים ה-Native VLAN ציר להיות זהה. בנוסף, כל Frame שמגיע למstag ללא Tag אוטומטית משוייך לו-Native VLAN. תכונה זו מאפשרת להתקנים שלא תומכים בפרטוקול 802.1Q, להיות שייכם לו-VLAN1.

הגדרת Trunk Port

```
Switch(config-if)#switchport mode trunk
```

```
Switch(config-if)#switchport trunk encapsulation {dot1q}
```

.Packet Tracer switchport trunk encapsulation dot1q לא נתמכת ב-

בדיקות ההגדרות

כדי לראות איזה Ports מוגדרים כ-Trunk

```
Switch#show interfaces trunk
```

חסימת vlans מסוימים במשיק Trunk

כברירת מחדל, משיק שמוגדר כ-Trunk מעביר את כל סוגיו ה- vlans (all).

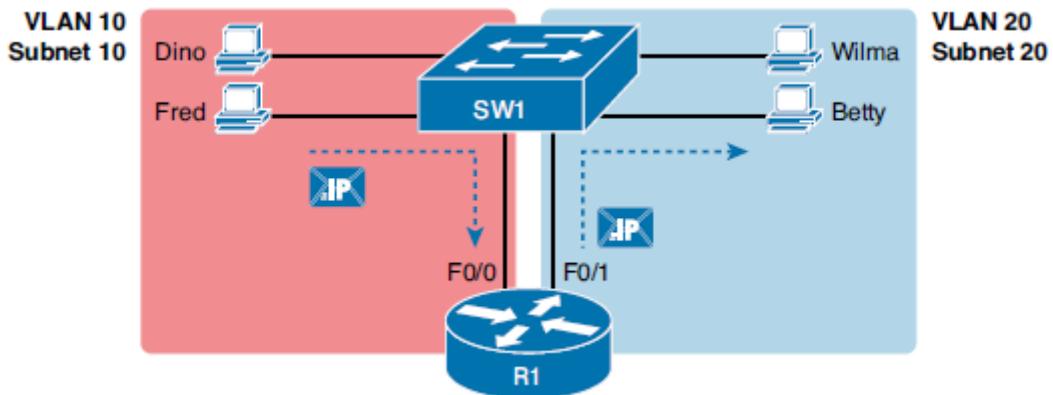
```
Switch(config-if)#switchport trunk allowed vlan {all/none/10,20/10-30/add 30/remove 30/except 30}
```

סיבה נוספת שיכולה למנוע מישיק Trunk להעביר Frames שישיכים לו – vlans מסוימים זה אם המstag מקבל Frame להעברה אבל הוא לא קיים במתג.

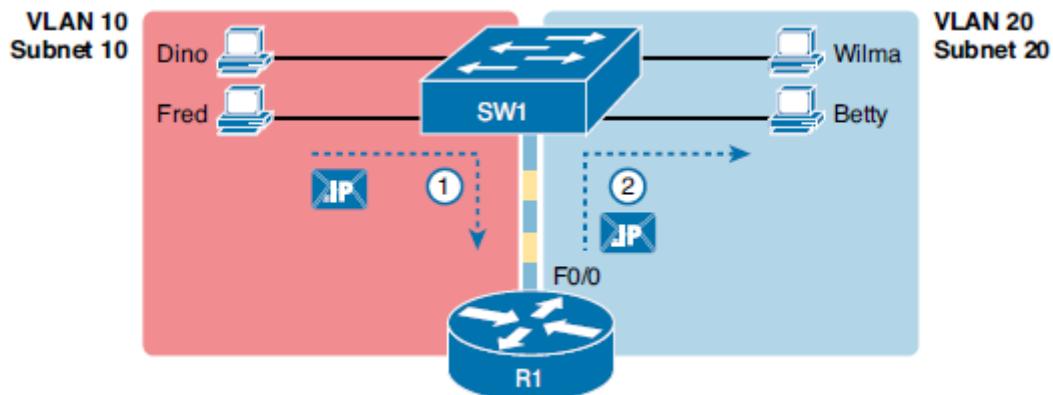
חיבור בין VLANs

כדי לאפשר תקשורת תקינה, יש לאפשר תקשורת בין VLANs שונים. לדוגמה לחבר בין השירותים לבין שאר המחשבים. כדי לחבר בין VLAN יש צורך בהתקן שעבוד בשכבה 3 (נתב או מtag שעבוד בשכבה 3). בעזרתו נתב ניתן לגורם לשני VLANs שונים, לתקשר אחד עם השני.

ניתוב בין שני VLANs באמצעות שני ממשקים בנתב



ניתוב בין שני VLANs באמצעות ממשק אחד בנתב (router-on-a-stick)



- אסור לחתת כתובת IP לממשק הראשי אלא רק לחתת ממשקים.
- חובה להפעיל את הממשק הראשי (NO SHUTDOWN)
- השימוש ב- ACL כדי לסנן תעבורת רשת.

במtag - צריך להפוך את היציאה שמחוברת לנtab ל- Trunk

Switch(config-if)#switchport mode trunk

בנתב - יש לחתת כמה כתובות IP לאוטו לממשק בנתב (subinterface)
Router(config)#interface fastEthernet 0/0.{vlan-id}
Router(config-subif)#encapsulation dot1Q {vlan-id}
Router(config-subif)#ip address {IP} {subnet mask}

VLAN Trunking Protocol (VTP)

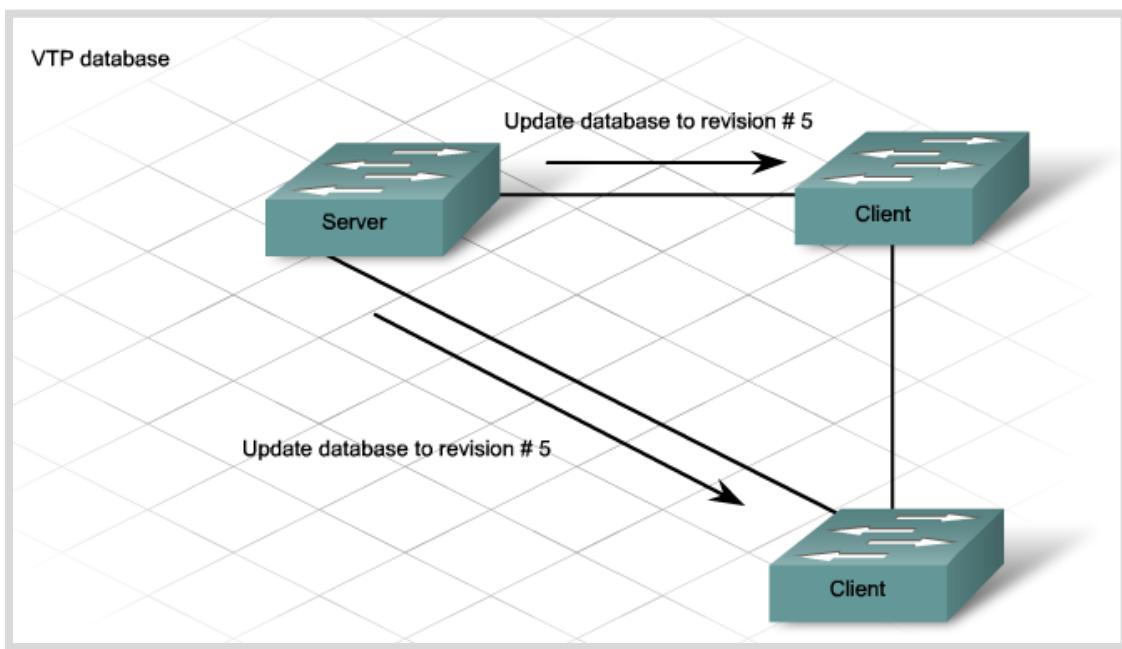
VTP זה פרוטוקול שעבוד בשכבה השנייה ומאפשר ניהול של VLANs ממוגן מרכז. נתבים לא מעבירים עדכנים של פרוטוקול זה.

VTP משפיע על תחום מוגדר (VTP Domain). לכל VTP Domain יש שם ייחודי. מתגים מחליפים בינם הודעות של VTP רק אם הם שייכים לאותו Domain. Trunk ports. עדכנים עוברים רק על

מתג שמוגדר לעבוד עם VTP, יכול לעבוד באחד משלשות מצביו העבודה הבאים:

- **Server** – מאפשר יצירה/מחיקה ו שינוי של VLANs ושולח עדכנים דרך יציאות של Trunk לכל המתגים שהם Clients שייכים לאותו Domain. כבירית מחדל, מתג מוגדר כ- Server.
- **Client** – מקבל עדכנים ומשנה הגדרות לפי העדכנים שmag'יעים מהשרת. מעביר עדכנים דרך יציאות של Trunk. לא ניתן ליצור VLAN מקומית.
- **Transparent** – לא משנה הגדרות אף מעביר עדכנים דרך יציאות של Trunk.

יש שני סוגי גרסאות של VTP, version 1 ו- 2. version 2 אין תאימות בין הגרסאות. כל המתגים שבאותו Domain חיברים לשימוש באותה גרסה של VTP. כל מתג שמקבל עדכן מהשרת בודק אם גרסת העדכן (revision) עדכנית יותר מהבסיס נתוניים שלו, ואם כן המתג מעדכן את בסיס הנתוניים שלו.



ל VTP יש שלוש סוגי הודעות:

- **Summary Advertisements**

מפרסם את שם ה- domain ו revision number כל חמישה דקות ובכל פעם שיש

שינוי. מtag ששייר ל- domain מקבל את ההודעה, ובודק אם הגרסה זהה יותר עדכנית. אם כן הוא מבקש עדכון יותר מפורט (advertisement request).

Advertisement Requests •

בקשת עדכון נשלחת כאשר: המtag אופס, שם ה Domain אליו שייר המtag שונה או כתוצאה מקבלת Summary Advertisements revision number גובה יותר מי זה של המtag.

Subset •

מכיל מידע מפורט לגבי השינויים. נשלח כתגובה לAdvertisement Requests.

VTP הגדרה

לא לשכוח שהמשתמשים שמחברים את המtags צריכים להיות מוגדרים כ- Trunk. מtags לומדים אוטומטית את ה- domain-name שמספרם המtag שמוגדר כ- server.

1. הגדרת המtag

```
Switch(config)#vtp mode {server/client/transparent}  
Switch(config)#vtp domain {domain-name}  
Switch(config)#vtp password {password}
```

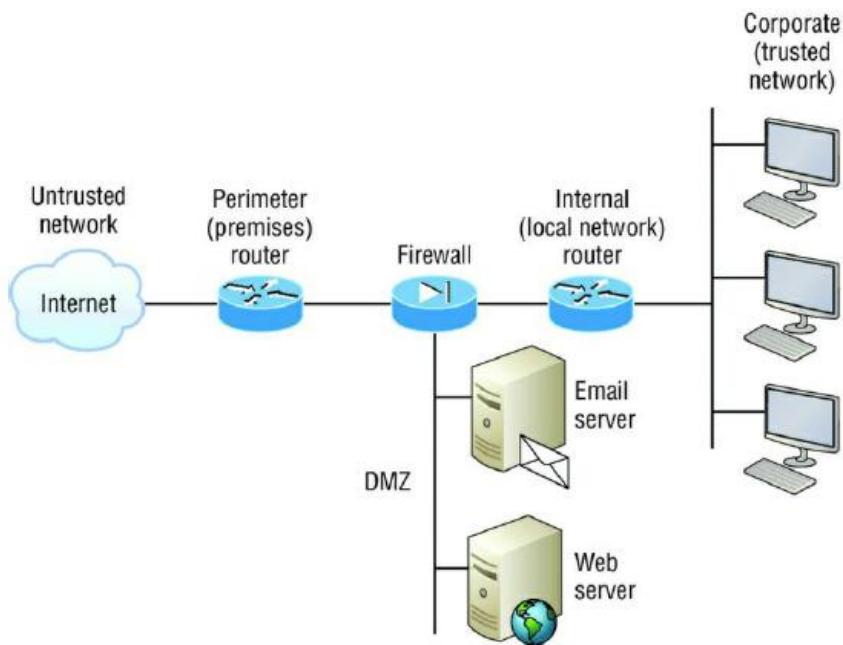
2. בדיקת ההגדרות

```
Switch#show vtp password  
Switch#show vtp status
```

. הערה: לא צריך להגדיר password vtp במtag שמוגדר כ- Transparent.

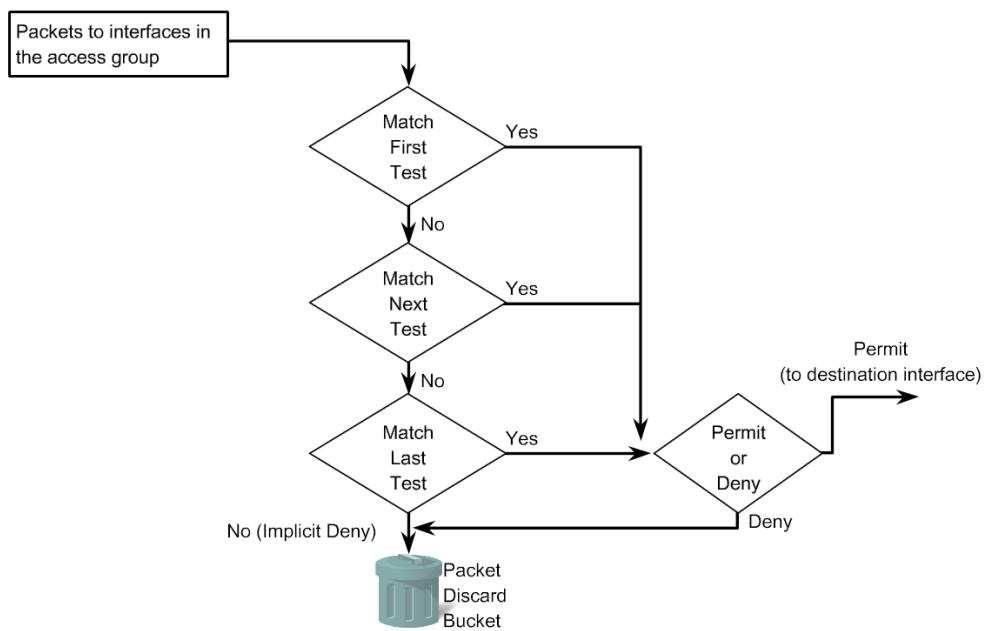
Access Control Lists (ACL)

ACL זו רשימת חוקים שבוצעים סיכון לubyteות הרשות וכן מגברת רמת האבטחה ומשתפרים ביצועי הרשות על ידי הפחיתה תעבורת רשות נוספת.
למעשה ACL מתקף כ- Packet Filtering Firewall



כיצד ACL עובד?

הביבלה שנכנתה לנתח נבדקת מול כל חוק החל מהחוק הראשון, כדי לגלוות לאיזה חוק היא תואמת. החביבלה ונחסמת או עוברת בהתאם לחוק הראשון שמתאים לה. אם אין חוק שתואם לחביבלה, החביבלה נזרקת וזאת בגלל שהחוק האחרון בכל ACL נוצר אוטומטית וחוסם הכל (implicit deny).



סוגי ACL

- **Standard ACL** - משתמש רק בכתובת המקור של החבילה כתנאי התאמה. לא ניתן להתייחס ל- port היעד.
- **Extended ACL** - משתמש בכתובת מקור/יעד, פרוטוקול, שכבה שלוש ומספר Port.
- **Named ACL** - זה ACL מסווג או Standard ACL אך גמיש יותר.

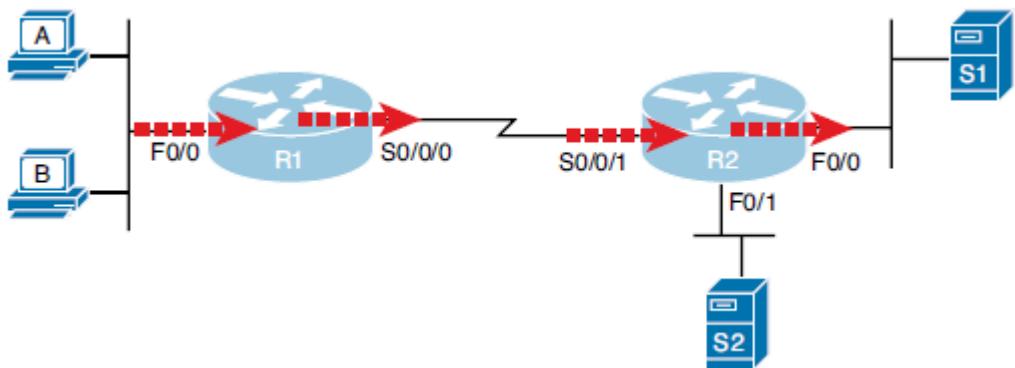
כיוון הבדיקה

ACL לא פעיל עד אשר הוא לא מוצמד לממשק מסוים.

בנוסף יש לציין את כיוון הבדיקה:

- **Inbound** - בדיקת תעבורת שנכנסת לממשק.
- **Outbound** - בדיקת תעבורת שיצאת מההממשק.

ניתן להצמיד לכל ממשק רק ACL אחד לכל כיוון.



כללים חשובים

- תעבורת שנוצרת על-ידי הנטב לא נחסמת.
- כדי ליעל את הבדיקה ולשחרר את המידע מהר,שים את ה חוקים שמתיחסים למრבית התעבורת הראשונית.
- לא ניתן לבצע שינויים ב- ACL אלה למחוק ולצורך חדש. Name ACL ניתן לערכיה. כדי להקל על העריכה, כתוב את חוקי ACL בכתבן ולאחר מכן, העתק אותו לנטב.

הכנון ACL

שימוש ACL דרוש תכנון, תכנון לא נכון או מיקום ACL בנטב או הגדזה לממשק לא נכון יכול לגרום לחסימת מידע חיווני או יצירה עומס בנטב.

תכנון נכון כולל את שלושת השלבים הבאים:

1. בדוק מהם דרישות הארגון. דרישות אלו יקבעו איזה סוג תעבורת יעבור/יחסם.
2. איזה סוג של ACL יתאים לדרישות (Extended ACL או Standard ACL).
3. באיזה נטב, ממשק וכיוון יפעל ACL.

Standard ACL

מתיחס רק לכתובת המקור ולכון נמקם אותו קרוב ליעד.

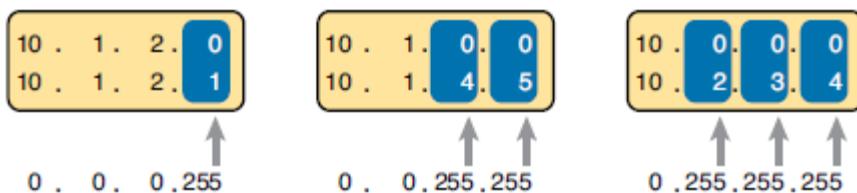
access-list {access-list-number} {deny|permit|remark} {source address} {source-wildcard}

מספר הרשימה יכול להיות 1-99 או 1300-1999.

Wildcard masks

משמש כתחליף ל- subnet mask.

0 מיצג bit שאסור לשנות (כתובת רשת) ו 1 מיצג bit שנותן לשנות (כתובת המחשב).



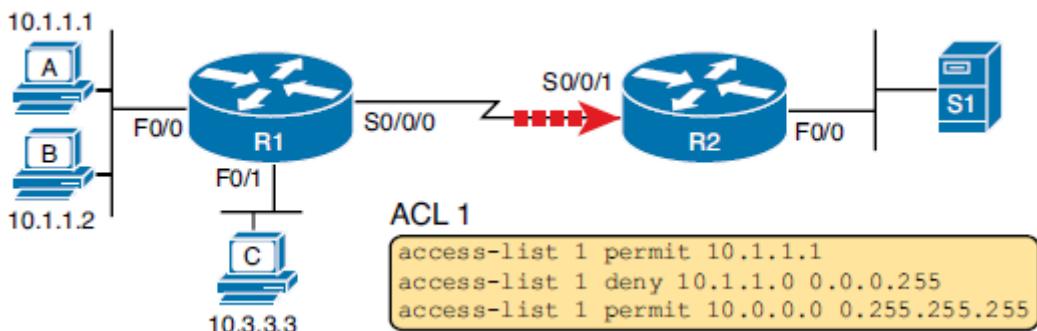
Host

במערכות ISO חדשות. ניתן לרשום את הכתובת IP ללא המילה host.

R1(config)#access-list 1 deny 192.168.15.99 **0.0.0.0** במקום
R1(config)#access-list 1 deny **host** 192.168.15.99 נרשם

Any

R1(config)#access-list 1 permit 0.0.0.0 255.255.255.255 במקום
R1(config)#access-list 1 permit **any** נרשם



Host A

S_IP = 10.1.1.1 →

✓ If Source = 10.1.1.1 Permit
If Source = 10.1.1.x Deny
If Source = 10.x.x.x Permit

Host B

S_IP = 10.1.1.2 →

✗ If Source = 10.1.1.1 Permit
✓ If Source = 10.1.1.x Deny
If Source = 10.x.x.x Permit

Host C

S_IP = 10.3.3.3 →

✗ If Source = 10.1.1.1 Permit
✗ If Source = 10.1.1.x Deny
✓ If Source = 10.x.x.x Permit

שיוך ACL לממשק

R1(config-if)#ip access-group {access-list-number} {in | out}

שליטה בגישה דרך VTY (Telnet/SSH)

ניתן לנצל בצורה ייעילה, למי תהיה גישה מרוחק לניהול התקן דרך Telnet או SSH. שימוש בשיטה זו בודק רק חבילות שמנסות להיכנס דרך vty ולכך לא מכבד על המערכת.

```
R1(config)#access-list {10} permit host {172.16.1.10}
R1(config)#line vty {0 4}
R1(config-line)#access-class {10} in
```

Monitoring ACL ACLs לראות פרוט של

```
Router#show access-list
```

```
Router#show access-lists
Standard IP access list 10
    deny host 172.17.0.1 (1 match(es))
    permit any (1 match(es))
```

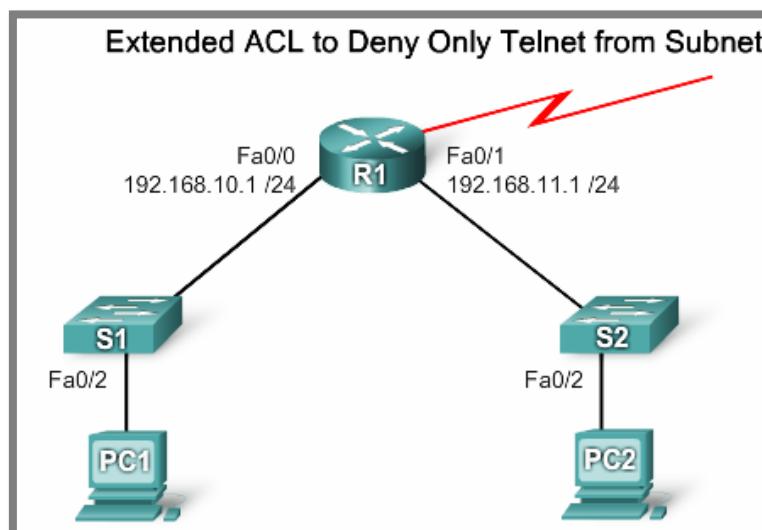
כדי לראות על איזה ממשך מוצמד ACL

```
Router#show ip interface
```

Extended ACL

אפשר/חומר לפיצוח מוקור/יעד ומספר Port. בדרך כלל נמקם Extended ACL קרוב למקור וכך מידע מיותר לא זורם ברשת. מספר הרשימה של ACL יכול להיות 100-199 או 2000-2699.

```
access-list access-list-number {deny | permit | remark} protocol source [source-wildcard]
[operator operand] [port port-number or name] destination [destination-wildcard] [operator
operand] [port port-number or name][established]
```



```
R1(config)#access-list 101 deny tcp 192.168.11.0 0.0.0.255 any eq 23
R1(config)#access-list 101 permit ip any any

R1(config)#interface Fa0/1
R1(config-if)#ip access-group 101 in
```

הוספה Remark

ניתן להוסיף שורות של הערות בכדי לגרום לכך ACL להראות מובן יותר.
.show running-config

```
R1(config)#access-list { 10 } remark { the remark }
```

הוספה Log

בסוף כל חוק ניתן להוסיף את המילה log וכן בכל פעם שהבילה תהיה תואמת לחוק תיווצר רשומה log.
זה מאפשר לעקב אחר הפעולות ברשת כולל ניסיונות לבצע פעולות אסורה.

Named ACL

MSG ACL או Extended ACL שמזוהה על-ידי שם ולא על-ידי מספר.

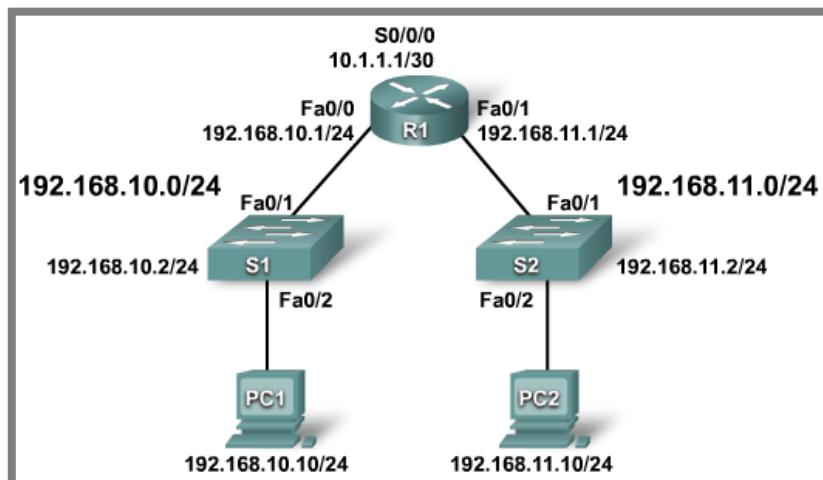
1. יצירת ACL Named ACL

```
Router(config)# ip access-list [standard | extended] name
```

2. יצירת חוקים

```
Router(config-std-nacl)# [permit | deny | remark] {source [source-wildcard]} [log]
```

3. שיקוף ACL למשק



```
R1(config)#ip access-list standard NO_ACCESS
R1(config-std-nacl)#deny host 192.168.11.10
R1(config-std-nacl)#permit 192.168.11.0 0.0.0.255
R1(config-std-nacl)#interface Fa0/0
R1(config-if)#ip access-group NO_ACCESS out
```

עריכת Named ACL

היתרון הגדול ביותר בשימוש ב- Named ACL, זה היכולת לבצע שינויים ב-ACL.

R1# **show access-lists** בדיקה:
Standard IP access list WEB SERVER

```
10 permit 192.168.10.10  
20 deny   192.168.10.0, wildcard bits 0.0.0.255  
30 deny   192.168.11.0, wildcard bits 0.0.0.255
```

R1(config)# **ip access-list standard WEB SERVER** עריכה:
R1(config-std-nacl)# **15 permit host 192.168.11.10**

R1# **sho access-lists** בדיקה:
Standard IP access list WEB SERVER

```
10 permit 192.168.10.10  
15 permit 192.168.11.10  
20 deny   192.168.10.0, wildcard bits 0.0.0.255  
30 deny   192.168.11.0, wildcard bits 0.0.0.255
```

Static Routing

מטרת הניתוב:

ניתוב זו פועלה שמאכזע בדרך כלל נתב. מטרת הניתוב היא:

- העברת packets בין רשתות שונות.
- מציאת הנתיב הטוב ביותר בדרך לעד.
- נתב לא מעביר שידור Broadcast domains ולכן מחלק את הרשת ל- Broadcast domains שונים.

מה זה טבלת ניתוב?

כדי לקבל החלטה, لأن לנtab החלטה, Packet, הנתיב משתמש בטבלת ניתוב. בטבלת ניתוב מופיעים כל הרשותות שהנתיב מכיר ומידע כיצד לנtab את ה- packets אליהם.

Router#show ip route

כדי לראות את טבלת הניתוב:

```
Router#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

C    10.0.0.0/8 is directly connected, FastEthernet0/0
C    176.16.0.0/16 is directly connected, FastEthernet0/1
```

מטרת ניתוב סטטי:

נתב לא מכיר את כל הרשותות שקיימות באינטראנט או ב- LAN, אלא רק רשותות שמחוברות אליו פיסית. בעזרת ניתוב סטטי ניתן ללמד נתב, נתבים לרשותות חדשות.

יכיז לנtab מחייב לנtab Packet שmag'ע אליו?

- תחילה נבדקת טבלת הניתוב. אם קיימים נתיב לעד, הנתיב משתמש בנתיב זה.
- אם העד לא מופיע בטבלת ניתוב, הוא משתמש ב- Default Route שזה בדרך כלל הנתיב הבא בדרך לספק שירותי (ISP).
- לבסוף, אם גם אין Default Route, ה- Packet נזוק.

מה זה ניתוב דינמי?

זו שיטה שמאפשרת לנtabים ללמידה את טופולוגיה הרשת בצורה אוטומטית. פרוטוקולי ניתוב שמוגדרים בנtabים, מתחברים אחד עם השני ומלמדים אחד את השני את מבנה הרשת .

יתרונות ניתוב סטטי:

- נתבים סטטיים לא מפורטים ברשף ולכן ניתוב סטטי בטוח יותר.

- ניתוב סטטי לא מפרסם נתיבים (קורסיה בניתוב דינאמי) ולכн לא מבוזר רוחב פס.
- ניתוב סטטי פחות מעmis על המעבד של הנtab ולכн ביצועי הנtab משתפרים.
- אם מוגדר בנtab ניתוב סטטי, הנתיבים בהם משתמש הנtab ידועים מראש.

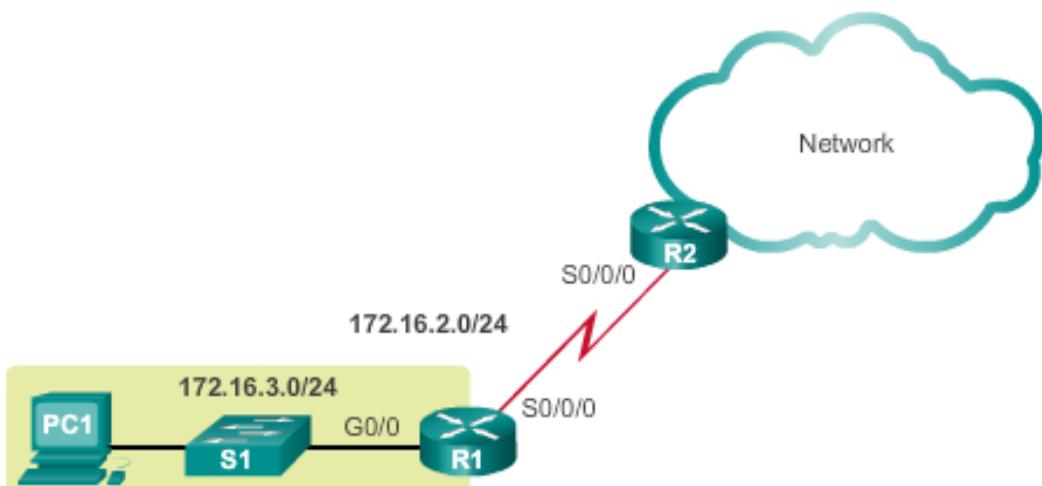
הסדרונות ניתוב סטטי:

- הגדרה ותחזוקה של נתיבים לוקחת זמן.
- בהגדרה של נתיבים סטטיים (במיוחד ברשנות גדוות) יש מקום לטעויות.
- כאשר יש שינוי בטופולוגיה הרשת, יש צורך בהתערבות ידנית של מנהל הרשת ולכн התחזוקה הופכת למסורבלת.
- כדי לישם ניתוב סטטי בצורה יעילה, נדרשת התמצאות מלאה בטופולוגיה הרשת.

ניתוב סטטי יעיל במיוחד ברשנות קטנות או רשות עם צורך ברמת אבטחה גבוהה.
כיום רוב הרשות משמשות בשילוב של ניתוב סטטי ודינאמי.

דוגמאות:

בנתב R1 נגידר את R2 כ- default static route כדי שככל תעבירת הרשת שיזאצא תגיע לנtab R2.
בנתב R2 נוצר רשומה ניתוב סטטי ש כדי להגיע לרשת 172.16.3.0/24 יש לשולח את המידע לנtab R1.

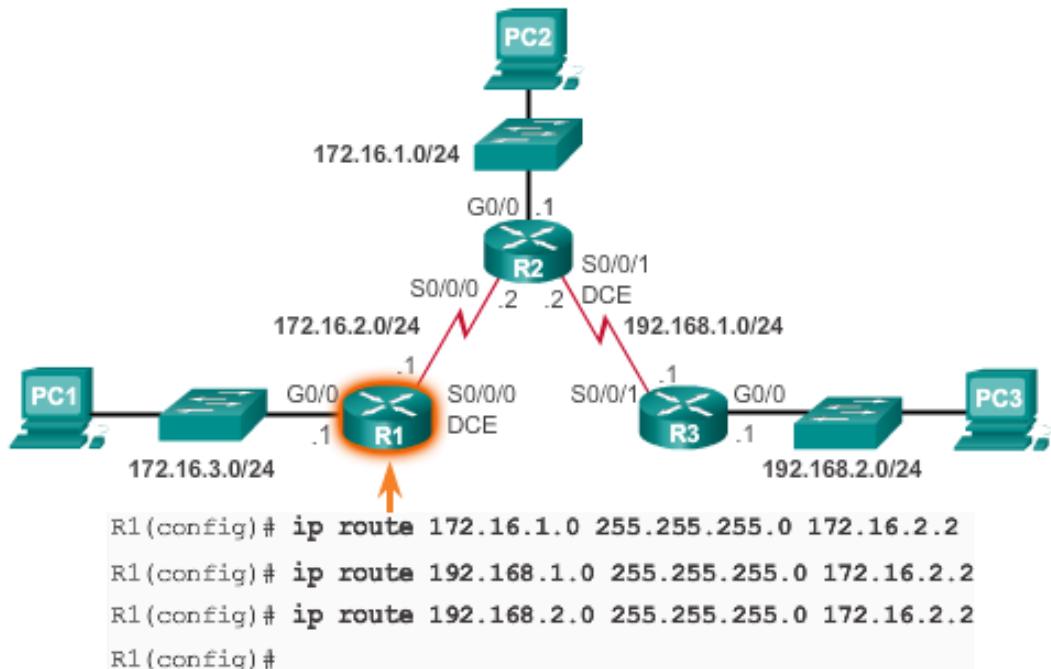


הגדרת ניתוב סטטי:

Router(config)#ip route {destination_network} {subnet_mask} {gateway_address}

Configuring Next Hop Static Routes on R1

דוגמא:

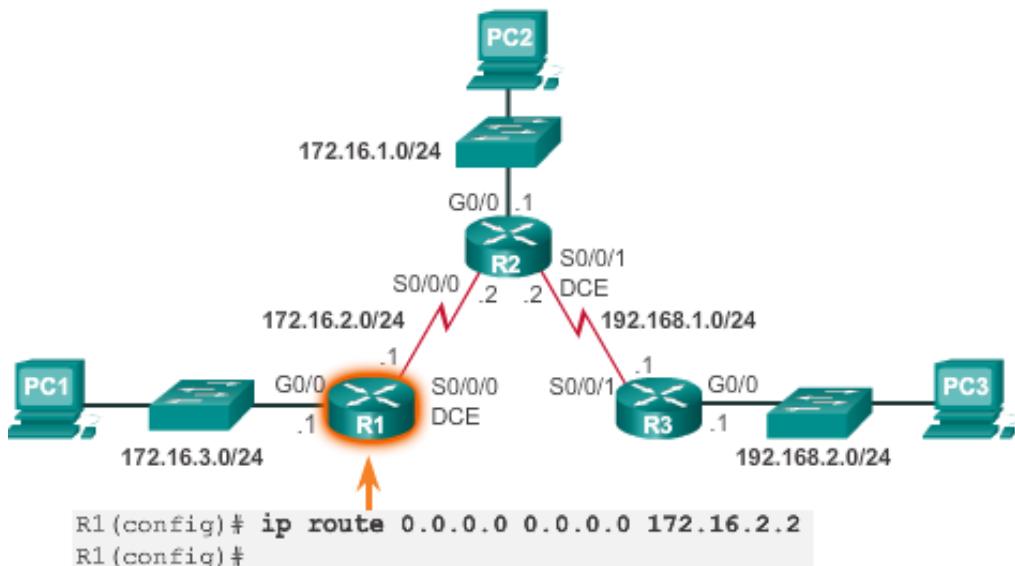


הגדרת Default Route

Router(config)#ip route 0.0.0.0 0.0.0.0 {gateway_address}

Configuring a Default Static Route

דוגמא:



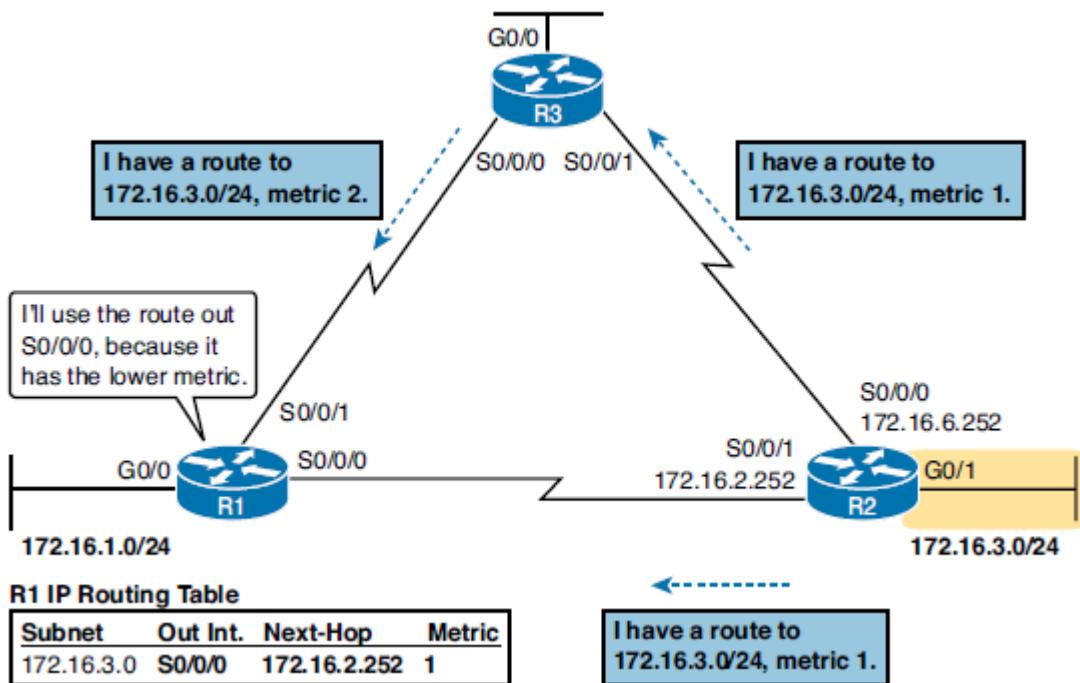
Routing Protocols

תפקיד פרוטוקול ניתוב:

- הנתב לומד מנתבים שכנים על קיומ נתבים חדשים ברשת.
- הנתב מפרסם לנטים השכנים את הנתבים שהוא מכיר.
- הנתב יודע לדרג את הנתבים לפי יעילות הנתיב (metric) וזאת כדי לבחור את הנתיב הטוב ביותר במידה יש יותר מנתיב אחד לעד.
- כאשר טופולוגיה הרשת משתנה, הנתב יודע לעדכן ולהתעדכן על השינוי.
- התהליך מהרגע שבו יש שינוי בטופולוגיה ועד לרגע שבו כל הנתבים למדו את השינוי, נקרא convergence.

דוגמאות:

בדוגמה זו, R2 מלמד את הרשות על קיומ רשות חדשה (172.16.3.0/24). R1 מכניס לטבלת הניתוב את הנתיב עם המטריק הטוב ביותר.



Metric

לכל נתיב קיים ערך שקבע את עדיפות הנתיב ביחס לנתבים אחרים. הערך נקרא Metric.

פרוטוקולי הניתוב משתמשים בשיטות שונות כדי לחשב את המטריק (Metric):

- Hop count – מספר הנתבים שחייב צריכה לעבור
- Bandwidth – רוחב פס של החיבור

Load – עומס העברת הנתונים על החיבור

Delay – זמן שלוקח לחבילה לצאת מהמסך (זמן תגובה).

Reliability – אמינות החיבור

פרוטוקולי ניתוב שעובדים בשיטת Distance Vector

פרוטוקולי ניתוב שעובדים בשיטה זו הם: RIP, EIGRP, BGP. בשיטה זו, הנטב מכיר רק את הנתבים השכנים והם מלמדים אותו את הנתבים לכל רשות ולכך הנטב לא מכיר את טופולוגיה הרשת המלאה. לעומת זאת הנטב לא מכיר נתבים מרוחקים אלא רק את הנתבים אליו.

פרוטוקולי ניתוב שעובדים בשיטת Link State

פרוטוקולי ניתוב שעובדים בשיטה זו הם: IS-IS, OSPF. בשיטה זו, כל נתב רואה את מפת הרשת המלאה (מפת הדרכים) ומתוך מפה זו, הנתב בונה לעצמו את טבלת ניתובו. הנתב מנהל מסד נתונים בו קיימים מידע רב על רשתות ונתבים מרוחקים ומה הדרך לתקשר איתה. החישון בשיטה זו, בכל פעם ש משתנה שינוי ברשת, נתב צריך לבצע הרבה חישובים וזה גורם לנטב להשתמש ביותר זמן מעבד וזכרן מאשר שיטה אחרת.

Administrative Distance (AD)

נתב יכול ללמוד נתבים בדרכים שונות.

כאשר בטבלת ניתוב קיימים מספר נתבים שונים ליעד מסוים אז הנתיב עם ה- Metric הנמוך ביותר נבחר אך כאשר הנתבים השונים נלמדו בדרכים שונות, לא ניתן להסתמך על ה- Metric זה הערך שקבע לאיזה שיטת לימוד יש את העדיפות הגבוהה ביותר.

Route Type	Administrative Distance
Connected	0
Static	1
BGP (external routes)	20
EIGRP (internal routes)	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP (external routes)	170
BGP (internal routes)	200
Unusable	255

תהליך בחירת הנתיב הטוב ביותר ביויתר

בהתליך בחירת הנתיב הטוב ביותר ביויתר נלקחים בחשבון הפרמטרים הבאים:

- כתובות הרשות המדויקת ביותר.
- לדוגמא אם נכנסת לנחבות חבילות עם כתובות יעד של 172.16.2.1, ובטבלת הנתוב קיימות שני רשותות: 172.16.2.0/16 ו 172.16.0.0/16 או הנתיב הנבחר הוא של רשת 172.16.2.0/24.
- Administrative Distance
- Metric

כדי לבדוק האם הנתיב משתמש בפרוטוקול נתוב ואיזה פרוטוקולי נתוב פועלים בנתיב, השתמש בפקודה:
Router#show ip protocols

Routing Information Protocol (RIP)

זהו פרוטוקול נתוב שעומד בשיטת Distance Vector. ה프וטוקול נitem על ידי רוב הנתבים ומתאים לרשותות קטנות.
RIPv1 נוצר בשנת 1988 והוא לא תומך ב- VLSM בגלל שהוא לא מפרסם בטבלת נתוב את ה- subnet mask. RIPv2 נוצר בשנת 1998 והוא תומך ב- VLSM.

לפרוטוקול המאפיינים הבאים:

- כל 30 שניות שולח את כל טבלת הנתוב ולא רק את העדכנים. כך גם הנתיב יודע האם הנתבים השכנים מתפקדים. אם במשך 180 שניות הנתיב לא מקבל את טבלת הנתוב מהשכן אז השכן נחשב לא זמני.
- בגלל שעדכון לא מגיע מידית אז כאשר טופולוגיה הרשת משתנה, לוקח זמן עד שכל הנתבים לומדים על השינויים.
- RIPv1 משדר את הטבלת נתוב ב- Broadcasts. RIPv2 משדר את הטבלת נתוב ב Multicasts לכתובת 224.0.0.9.
- RIP בוחר את הנתיב הטוב ביותר ביחס למרחק (הנתיב בו יש הכי פחות Hops) ולא תומך בנתיב בו יש יותר מי 15 Hops.
- RIPv2 תומך בנתיב בו יש יותר מי 15 Hops.
- RIPv2 תומך ב authentication ובהצפנה המידע, RIPv1 לא תומך ב authentication והצפנה.

RIP Configuration

כדי להפעיל את ה프וטוקול נתוב בפקודה:

Router(config)#router rip

כברירת מחדל RIPv1 פועל והוא לא תומך ב- VLSM ולכן כדי לעבוד עם RIPv2

כדי לעبور ל- RIPv2 נתוב בפקודה:

Router(config-router)#version 2

הפקודה network מגדירה:

- דרך איזה ממשקים תפורסם טבלת הנתוב.
- איזה רשותות RIP יפרסם דרך טבלת הנתוב.

בדוגמה, הנתיב יפרסם כל רשת שמתחליה ב- 192.168.1.16 ו 172.16.1.19. יש להגדיר כתובות רשת מלאה (classful).

```
Router(config-router)#network {172.16.0.0}
Router(config-router)#network {192.168.1.0}
```

לא לשכוח שלוקח לו RIP 30 שניות להתרען.

automatically summarize

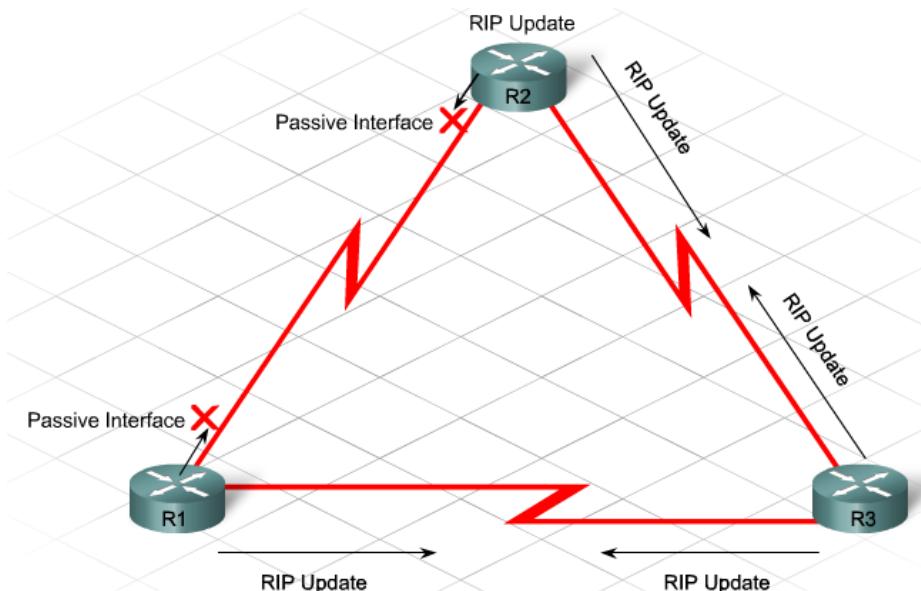
שני פרוטוקולי הניתוב (1 ו 2) מבצעים כבירית מחדל automatically summarize, כלומר, כלומר מצמצמים כמה תתי רשותות לרשת אחת בגבולות A, B או C. ניתן לבטל את ה指挥ם רק ב 2 RIPv2 באמצעות הפקודה:

```
Router(config-router)#no auto-summary
```

passive-interface

פרוסום טבלת הניתוב דרך כל הממשקים יכול לגרום לביעות אבטחה ולתעבורה רשת מיותרת. באמצעות פקודה זו ניתן לא לפרסם על ממשק מסוים:

```
Router(config-router)#passive-interface interface-type interface-number
```



בדיקות הנטוֹרָנוֹת

כדי לראות את טבלת הניתוב:

```
Router#show ip route
```

כדי לראות פרטים על פרוטוקול הניתוב שמוגדרים בנטב:

```
Router#show ip protocols
```

כדי לראות בזמן אמת את הפרוסומים שהנתב שולח והעדכנים שהנתב מקבל:
הפקודה debug mccivid על פעילות הנתב ולכן יש להשתמש בה לתקופה קצרה

```
Router# debug ip rip
```

ביטול הפקודה

```
Router# no debug ip rip
```

Subnetting IP Networks

חלוקת לחתמי רשתות (Subnetting)

לפעמים יש צורך לחלק את הרשות לרשתות משנה (subnetworks). רשתות משנה נוצרות על-ידי לקיחת מספר ביטים מ- Host ID והפיכתם לחלק .Network ID

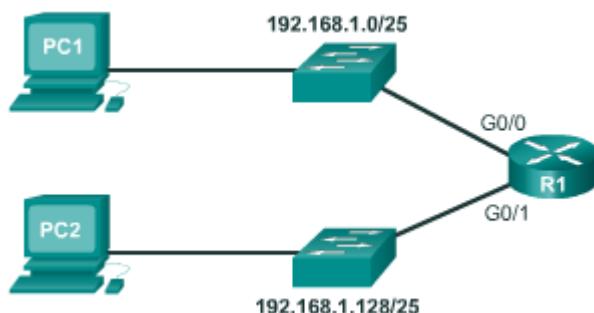
הכתובות רשות המקורית

Address	192	168	1	0000	0000
Mask	255	255	255	0000	0000
Network Portion					Host Portion

חלוקת הכתובות רשות לשני חתמי רשתות על-ידי לקיחת בית אחד מהכתובות רשות

Net 0	192.	168.	1.	0	000	0000	Network: 192.168.1.0/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128
Net 1	192.	168.	1.	1	000	0000	Network: 192.168.1.128/25
Mask	255.	255.	255.	1	000	0000	Mask: 255.255.255.128

כל כתובת רשות משמשת מקטע רשות שונה



פרוטוקול כתובות: רשות, ראשונה, אחרונה, שידור לכל

Address Range for 192.168.1.0/25 Subnet

Network Address

192. 168. 1. 0 000 0000 = 192.168.1.0

First Host Address

192. 168. 1. 0 000 0001 = 192.168.1.1

Last Host Address

192. 168. 1. 0 111 1110 = 192.168.1.126

Broadcast Address

192. 168. 1. 0 111 1111 = 192.168.1.127

Address Range for 192.168.1.128/25 Subnet

Network Address

192. 168. 1. 1 000 0000 = 192.168.1.128

First Host Address

192. 168. 1. 1 000 0001 = 192.168.1.129

Last Host Address

192. 168. 1. 1 111 1110 = 192.168.1.254

Broadcast Address

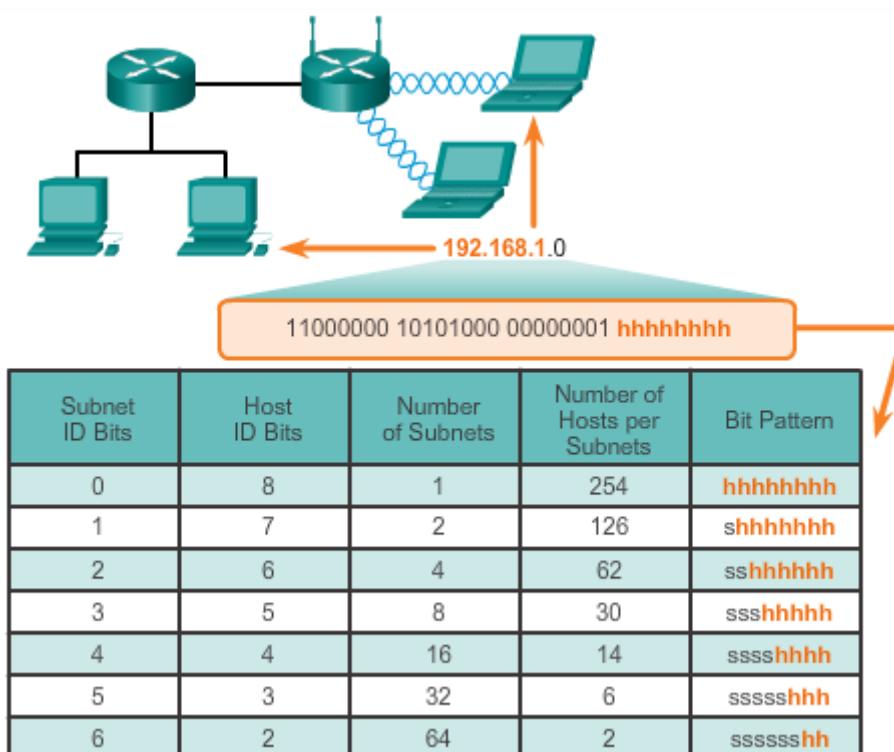
192. 168. 1. 1 111 1111 = 192.168.1.255

чисוב לפי דרישת - מספר תת-רשתות:
 כדי לדעת כמה ביטים יש לנקח משדה ה- Host כדי לקבל מספר מסוים של תת-רשתות, נשתמש בנוסחה הבאה: **מספר תת-רשת = 2^n** .
 ח מיצג את כמות הביטים ב- ID Subnet. המספר שמתקיים יהיה מספר משדה ה- Network Bits שעוביים משדה ה- Host לשדה ה- Network.

чисוב לפי דרישת - מספר כתובות IP בכל תת-רשת:
 כדי לדעת כמה ביטים צריכים להישאר בשדה ה- Host, כדי לקבל מספר מסוים של כתובות בכל תת-רשת, נשתמש בנוסחה הבאה: **מספר כתובות IP = 2^{n-2}** .
 ח מיצג את כמות הביטים ב- ID Host. אנו מחסרים 2, מכיוון ששתי כתובות שמורות ולא ניתן להקצתם למחשבים: כתובת הרשת וכתובת שידור לכל.

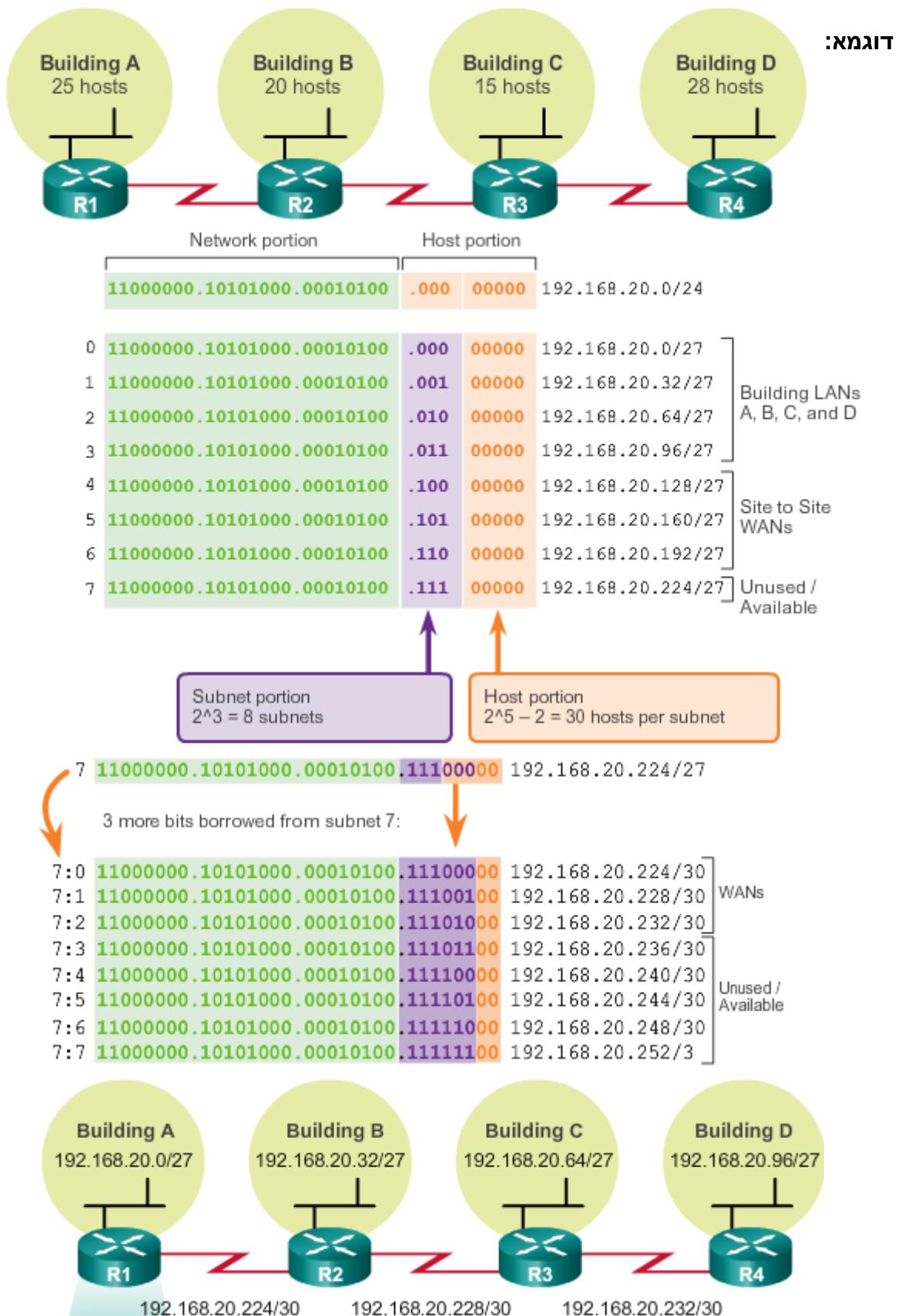
Network portion	Host portion	
10101100 . 00010100 . 0000000	00 . 00000000	172.16.0.0/22
		10 host bits $2^{10} - 2 = 1,022$ hosts

בדוגמה ניתן לראות את היחס בין כמות הביטים בשדה הרשת לבין כמות תת-רשתות. וגם את יחס בין כמות הביטים בשדה ה- Host לכמות כתובות IP זמינים.



VLSM (Variable Length Subnet Masking)

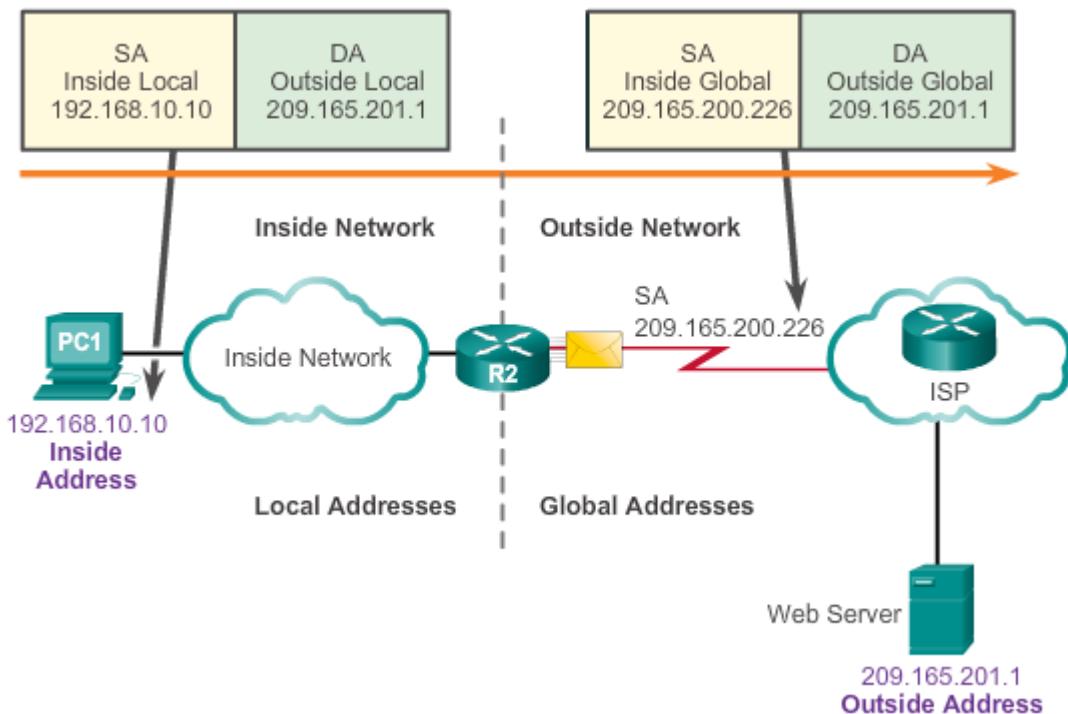
בחלוקת לחתני רשתות לכל תת רשת יש את אתה כמות של כתובות זו.
לא תמיד זה משקף את המציאות, וכתובות זו מתbezזות.
VLSM זו שיטה שבה אנו מחלקים את הרשת לחתני רשתות באורך משתנה.



Type of NAT Addresses

NAT Terminology

- כתובת המקור לפניו התרגום (כתובת פרטית).
- כתובת המקור לאחר התרגום (כתובת ציבורית).
- כתובת היעד (כתובת ציבורית).
- כתובת היעד (כתובת ציבורית).



NAT Overload (PAT)

מפה כתובות פרטיות רבות, לכתובת ציבורית אחת.

הכתובות לתרגום -	Router(config-if)#ip nat outside	הגדרת המשק החיצוני	הגדרת שמייעדות
	Router(config-if)#ip nat inside	הגדרת המשק הפנימי	

צירת access-list שגדיר את הכתובות הפנימיות

```
Router(config)#access-list {acl-number} permit {source} {source-wildcard}
```

הגדרת התרגום – שירך access-list לממשק החיצוני לצורך התרגום

```
Router(config)#ip nat inside source list {acl-number} interface {interface}  
overload
```

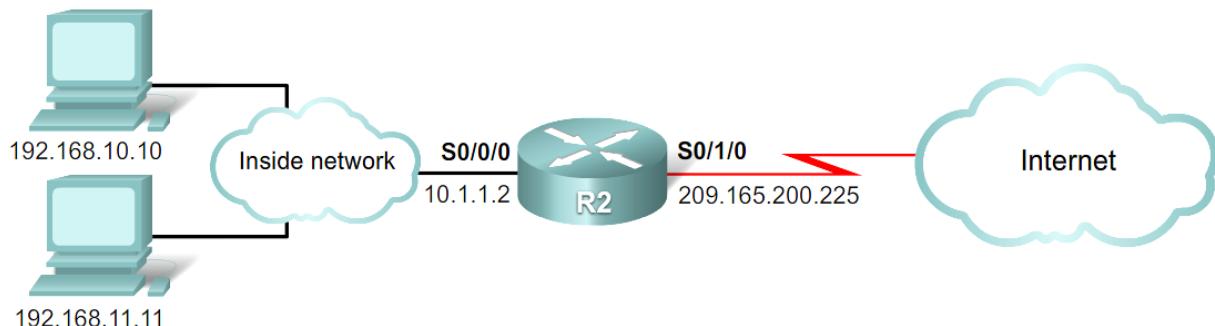
צפיה בטבלת התרגומים

```
Router#show ip nat translations
```

```
R2#show ip nat translations  
Pro Inside global           Inside local          Outside local        Outside global  
tcp 209.165.200.225:16642  192.168.10.10:16642  209.165.200.254:80  209.165.200.254:80  
tcp 209.165.200.225:62452  192.168.11.10:62452  209.165.200.254:80  209.165.200.254:80
```

דוגמה

כתובות פרטיות ברשת 192.168.0.0/16 יתורגמו לכתובת הציבורית 209.165.200.225



```
access-list 1 permit 192.168.0.0 0.0.255.255  
!Defines which addresses are eligible to be translated  
ip nat inside source list 1 interface serial 0/1/0 overload  
!Identifies the outside interface Serial 0/1/0 as the inside global address to be overloaded  
interface serial 0/0/0  
    ip nat inside  
!Identifies Serial 0/0/0 as an inside NAT interface.  
interface serial 0/1/0  
    ip nat outside  
!Identifies Serial 0/1/0 as an inside NAT interface.
```

Static NAT

מפה כתובה פרטית אחת, לכתובה ציבורית אחת.

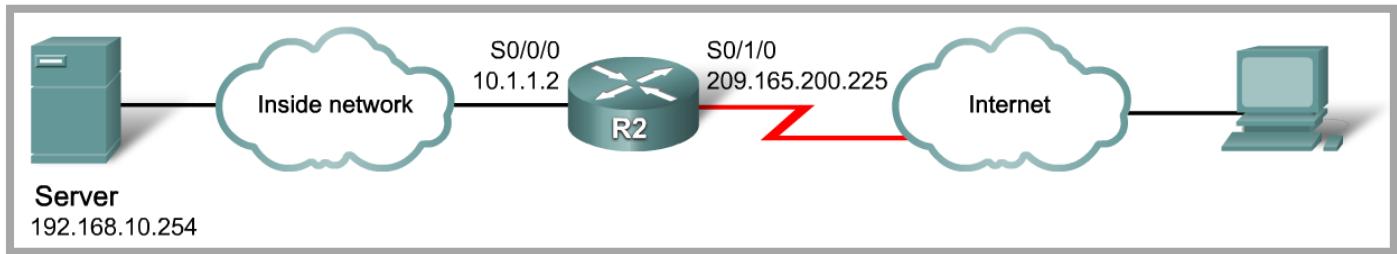
Router(config-if)#ip nat outside	הגדרת המmeshק החיצוני
Router(config-if)#ip nat inside	הגדרת המmeshק הפנימי

הגדרת התרגום

Router(config)#ip nat inside source static {local-ip} {global-ip}

דוגמה

הכתובת הפרטית 192.168.10.254 תתרגם תמיד לכתובת הציבורית 209.165.200.254



```
ip nat inside source static 192.168.10.254 209.165.200.254
```

!Establishes static translation between an inside local address and an inside global address.

```
interface serial 0/0/0
```

```
ip nat inside
```

!Identifies Serial 0/1/0 as an outside NAT interface.

```
interface serial 0/1/0
```

```
ip nat outside
```

!Identifies Serial 0/1/0 as an outside NAT interface.

Port Forward

כדי להסוך כתובות IP ציבוריות ניתן למפות את הכתובת הציבורית של הנטב ל- port היעד במחשב בתוקף הרשת וכך הנטב ידע למי צריך להעביר את ה-.Pakets

Router(config)#ip nat inside source static {local-ip} {port} {global-ip} {port}

Dynamic NAT

אנו משתמש בשיטה זו במקרה בו כמות ה- Connections יכולה להיות גדולה מ- 60,000. ממנה כתובות פרטיות, לכתובות ציבוריות מתוך מאגר כתובות ציבוריות.

Router(config-if)#ip nat outside	הגדרת הממשק החיצוני
Router(config-if)#ip nat inside	הגדרת הממשק הפנימי

הגדרת מאגר כתובות ציבוריות

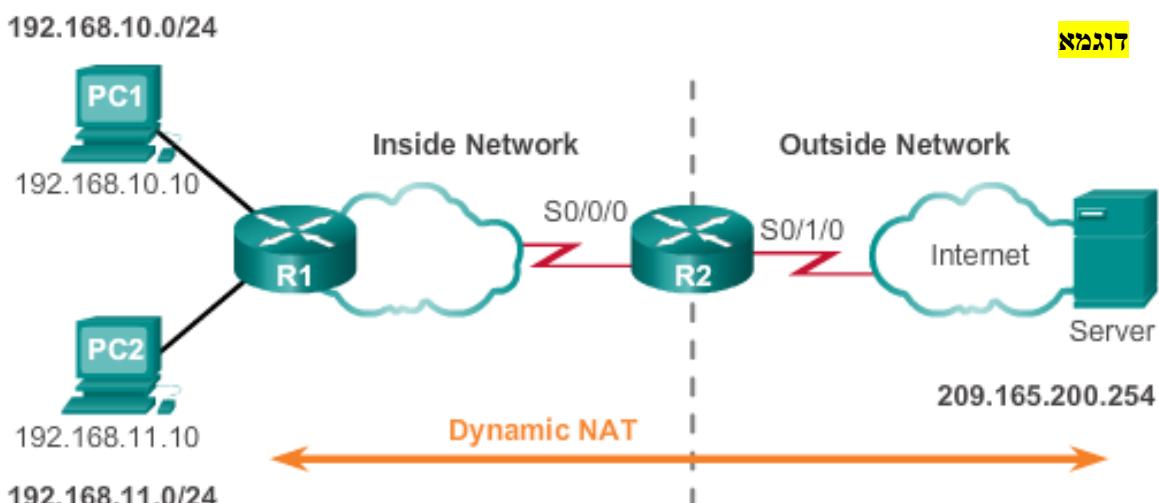
Router(config)#ip nat pool {pool name} {starting IP} {ending IP} {netmask}

הגדרת הכתובות שמיועדות לתרגום - יצירת access-list שמנדר את הכתובות הפנימיות

Router(config)#access-list {acl-number} permit {source} {source-wildcard}

הגדרת התרגום – שירט למאגר הכתובות הציבורית לצורך התרגום.

Router(config)#ip nat inside source list {acl-number} pool {pool name}



Defines a pool of public IPv4 addresses under the pool name NAT-POOL1.

R2 (config) # ip nat pool NAT-POOL1 209.165.200.226

209.165.200.240 netmask 255.255.255.224

Defines which addresses are eligible to be translated.

R2 (config) # access-list 1 permit 192.168.0.0 0.0.255.255

Binds NAT-POOL1 with ACL 1.

R2 (config) # ip nat inside source list 1 pool NAT-POOL1

Identifies interface serial 0/0/0 as an inside NAT interface.

R2 (config) # interface Serial0/0/0

R2 (config-if) # ip nat inside

Identifies interface serial 0/1/0 as an outside NAT interface.

R2 (config) # interface Serial0/1/0

R2 (config-if) # ip nat outside

Internet Protocol Version 6 (IPv6)

אייזה בעיות יש IPv4?

עם הזמן, יותר התקנים מתחברים לאינטרנט (טלפון חכם, טלויזיה, מדפסת...). כדי לחבר את כלם לאינטרנט, נדרשת כמות גודלה מאוד של כתובות IP ציבוריות. כמוות הכתובות הציבוריות מסוג IPv4 (אתם ניתן לתקשר באינטרנט) הולכת ואוזלת. השימוש ב- VLSM ו-NAT/PAT מאפשר לחסוך בכתובות IP אך יוצר בעיות מסוימות.

יתרונות IPv6

- כמויות "אין סופית" של כתובות IP

כתובת מסוג IPv6 באורך 128bit לעומת כתובות מסוג IPv4 שהיא באורך 32bit.

Version	Supported Addresses
IPv4	4,294,967,296
IPv6	340,282,366,920,938,463,463,374,607,431,768,211,456

כל אדם בכל רחבי הארץ יכול לקבל כמות זהה לכל הכתובות הציבוריות שקיימות ב- IPv4 וכן IPv6 מבטל את הצורך בשימוש ב- NAT או PAT.

קבוע של Subnet mask 64bit

כלומר ממחצית מהכתובת IP היא כתובת רשות.

ניתן לשנות את ה- subnet mask אך בדרך כלל אין בכך צורך.

אין שידור מסוג Broadcast Multicast

עושה את העבודה במקום.

תמייה מובנת ב- IPsec

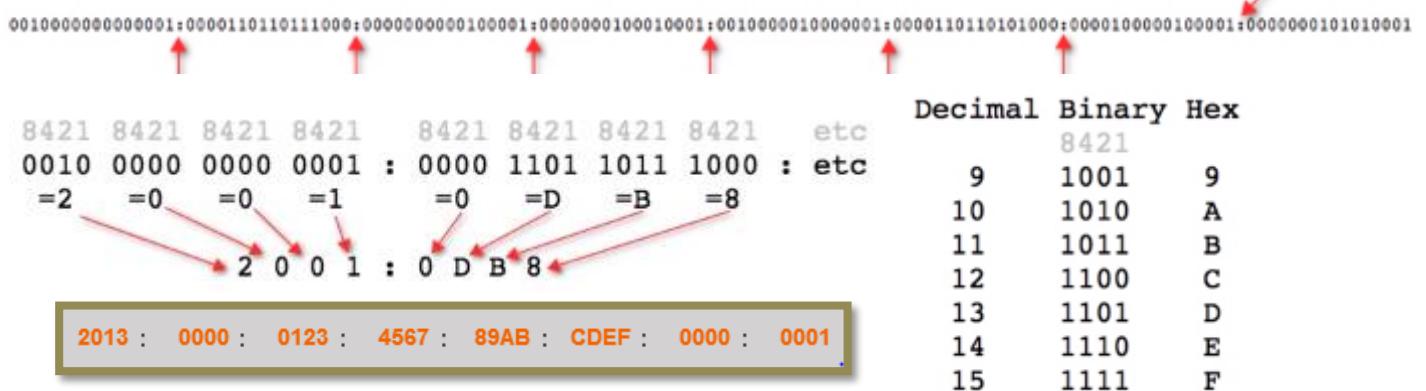
モבנה בכתובות מסוג IPv6, וכן שימוש ב- IPsec פשטוט יותר ותעובה הרישת בטוחה נוספת.

מבנה כתובת IPv6

כתובת IPv6 מורכבת מ- 128Bit

מתאימים נוחות הכתובות מחולקת לשםונה קבוצות מספרים (16Bit כל קבוצה).

הכתובת מתורגם למספר הקסדצימלי. בדרך כלל לא נכתבת הכתובת ביבינארית.



כתיבת הכתובת בצורה מקוצרת (צמצום הכתובת)

קיימים שני חוקים שמאפשרים למצוין את הכתובת לכתובת קצרה יותר.

חוק ראשון: ניתן להוריד אפסים בתחילת כל מספר.

2001:0DB8:0000:1111:0000:0000:0000:0200
2001: DB8: 0:1111: 0: 0: 0: 200
FE80:0000:0000:0000:0123:4567:89AB:CDEF
FE80: 0: 0: 0: 123:4567:89AB:CDEF

חוק שני: ניתן להחליף אפסים רצופים בסימן :: (רק פעם אחת בכל כתובות)

Preferred	2001:0DB8:0000:0000:ABCD:0000:0000:0100
No leading 0s	2001: DB8: 0: 0:ABCD: 0: 0: 100
Compressed	2001:DB8::ABCD:0:0:100
or	
Compressed	2001:DB8:0:0:ABCD::100

Only one :: may be used.

תרגילים:
מצוין את הכתובות.

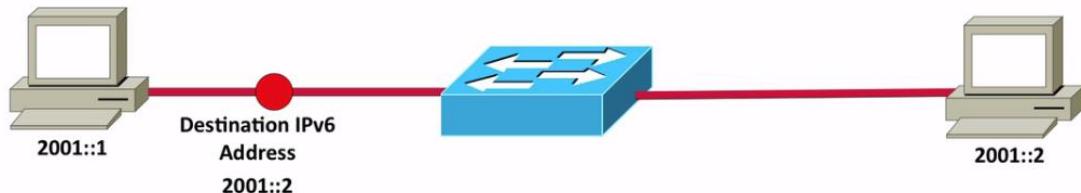
2001 : 0000 : 0DB8 : 1111 : 0000 : 0000 : 0000 : 0200
2013 : 0000 : 0123 : 4567 : 89AB : CDEF : 0000 : 0001
1111 : 0000 : 0000 : 0000 : 0000 : 0000 : 0101 : 1111
0000 : 0000 : 0000 : 1234 : 6678 : 9101 : 0000 : 34AB

פואגי פטיפובות

ב- IPv6 ישנן שלושה סוגי כתובות. לא קיימות יותר כתובות מסוג Broadcast.

Unicast

תקשרות מסוג אחד ל- אחד, כמו ב- IPv4.

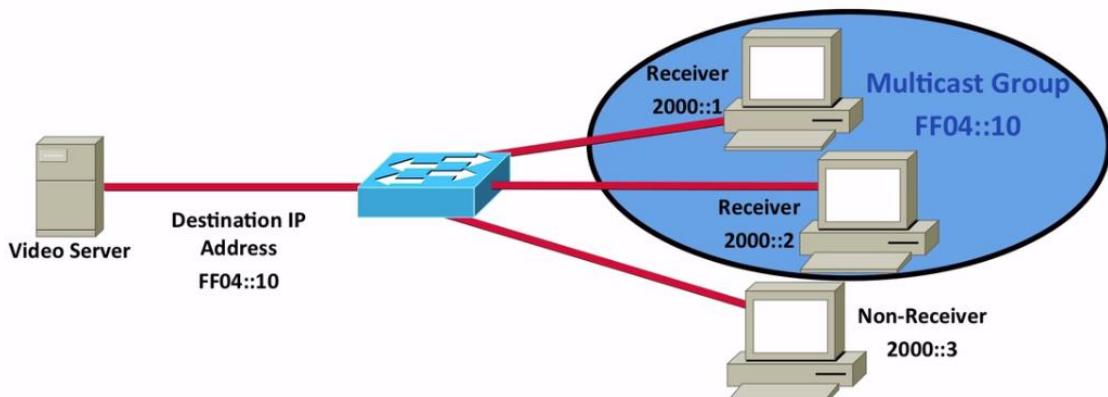


(FF00::/8) Multicast

זו תקשורת אחד לרבים, כמו ב- IPv4. Multicast מחליף את התקשרות מסוג Broadcast.

ישנן כתובות שמורות כמו:

- FF02::1 – כל התחקים
- FF02::2 – כל הנتابים

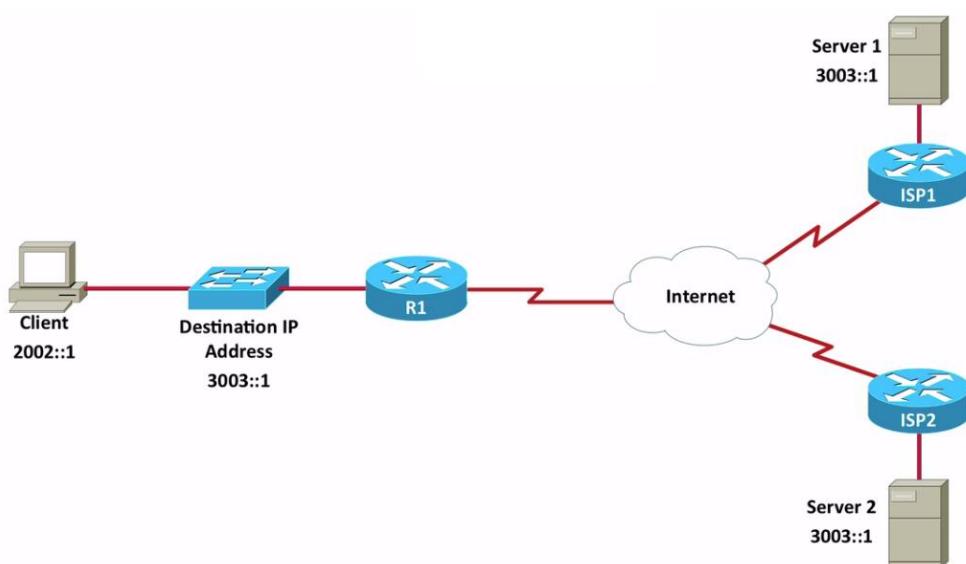


(one to nearest) Anycast

זו צורת שידור מסוג אחד לקבוצה אבל המידע לא יגיע לכל הקבוצה אלא לשרת הקרוב ביותר (קרוב לפי החלטת הנtab).

הכתובות מוגדרות רק בנtab ולא בשרת וכותבת אחת יכולה ליתג יותר מהtaskן אחד.

בנוסף, כתובת מסוג Anycast אף פעם לא תשמש ככתובת מקור.

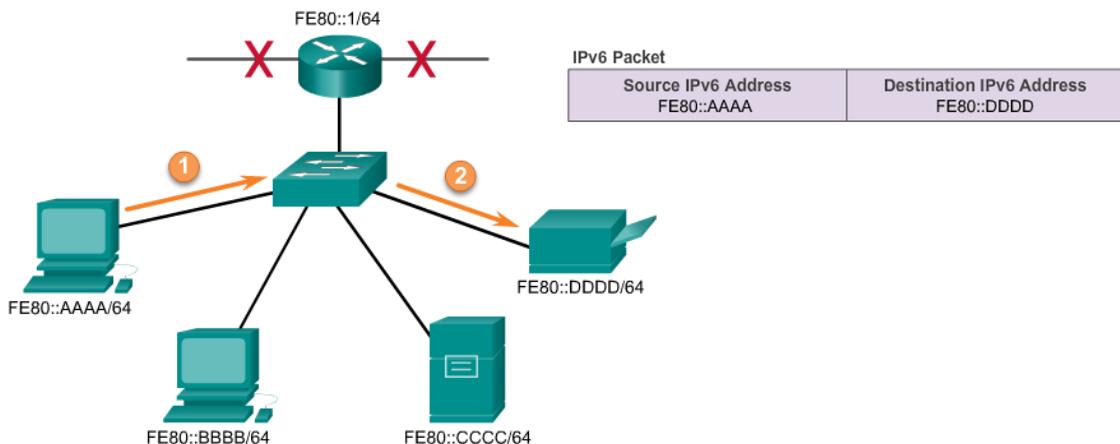


פונגי Unicast Addresses

חשוב לציין, לכל כרטיס רשת שמשתמש ב- IPv6 יהיה בדרך כלל כמה כתובות IP.

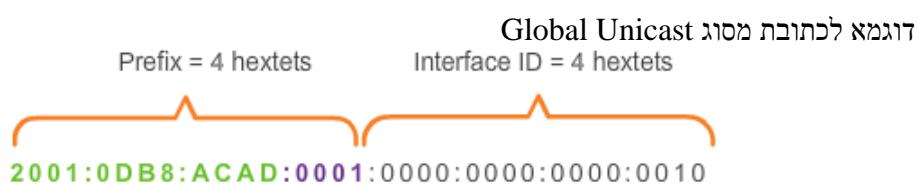
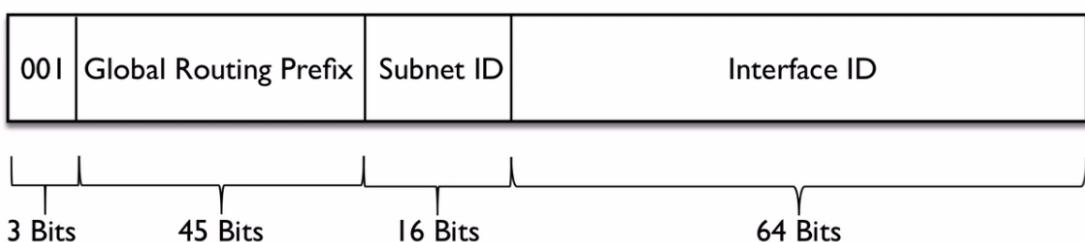
(FE80::/10) Link-Local

לכל כרטיס רשת שתווך ב- IPv6 חייבת להיות כתובת מסווג Link-Local (זומה ל- APIPA) ולכן משמשת לתקשורת רק במרחב Layer 2 .Domain



(2000::/3) Unique Global

זו כתובת ייחודית באינטרנט, כמו כתובת ציבורית ב- IPv4. הספק שירות מספק לlokה כתובת עם 48 Network ID /48, וזה משאיר לлокה 16bits כדי ליצור תתי רשתות.



(FC00::/7) Unique local

זו כתובת שלא ניתן לגלוש אותה באינטרנט, בדומה לכתובת פרטית ב- IPv4.

(::1) Loopback

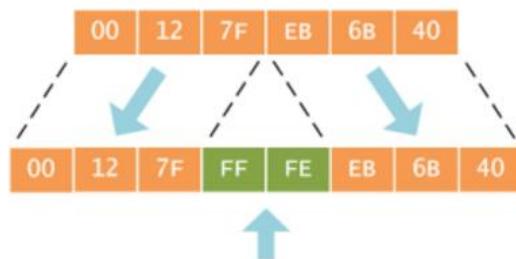
כמו 127.0.0.1

Extended Unique Identifier (EUI-64)

EUI-64 מאפשר להתקן לחת לעצמו כתובת (Unicast Global Unique local, Link-Local) או כתובת הפיסית של הכרטיס רשות כדי לקבוע את ה-ID Interface.

שלב ראשון:

כתובת פיסית באורך 48bit וANO צריכים כתובת באורך של 64bit, כלומר הסדרים לנו 16bit. לכן, נחלק את הכתובת הפיסית לשנים ונכנסים את הערך FFFE בין שני חלקים הכתובת הפיזית.



שלב שני:

7 bit בכתובת מזזה את הכתובת ככתובת מסוג (U/L). universal/local (U/L). בכל מקרה יש להחליף את ערך ה bit מי 0 ל 1 או מי 1 ל 0.



דוגמאות:

כתובת רשת: 2001:0db8:0:1::

MAC address: 0090:2716:fd0f

IPv6 EUI-64 address: 2001:0db8:0:1:0290:27ff:fe16:fd0f

MAC address: aa12:bcfc:1234

IPv6 EUI-64 address: 2001:0db8:0:1:a812:bcff:febc:1234

MAC address: 0c0c:dede:1234

IPv6 EUI-64 address: 2001:0db8:0:1:0e0c:deff:fede:1234

MAC address: 0b34:ba12:1234

IPv6 EUI-64 address: 2001:0db8:0:1:0934:baff:fe12:1234

Neighbor Discovery Protocol (NDP)

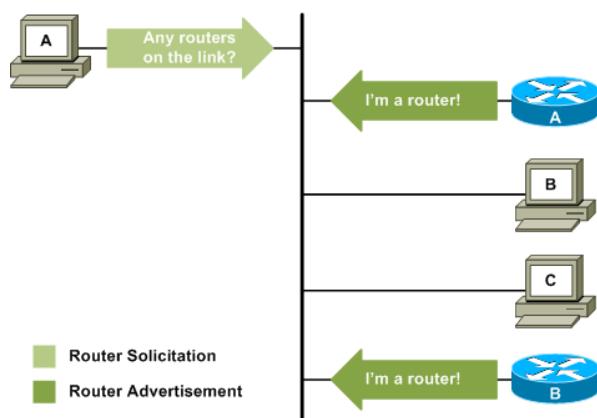
ב-IPv4, מחשב משתמש בפרוטוקול ARP ובשידור Broadcast כדי לגלות איזה התקנים מחוברים לרשת. ב-IPv6 אין ARP ואין Broadcast. ב-IPv6 נעשה שימוש בפרוטוקול ICMPv6 בהודעות Shnkeroutes לשידור מסוג Neighbor Discovery.

Router Solicitation (RS)

כאשר מחשב מתחבר לרשת, הוא מhapus איזה נתב פועל כדי לקבל ממנו פרטם על הרשת. הוא שולח הודעה מסוג RS ב- Multicast לכתובת 2::FF02. כל הנתבים מקבלים את ההודעה הזאת.

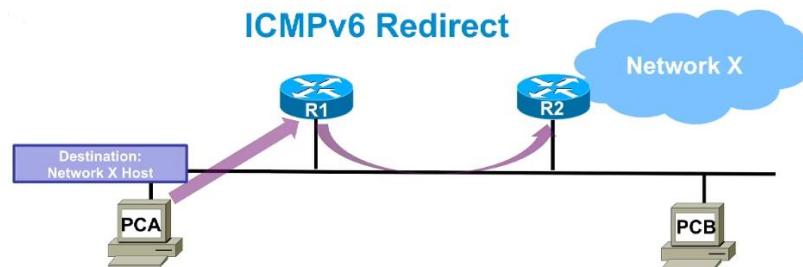
Router Advertisement (RA)

כאשר הנתב מקבל את ההודעה, הוא שולח בתגובה הודעה מסוג RA. אחד התגובה שולח הנתב ב- Unicast לשלו. הנתב מלמד את הלקוח את הכתובת רשת שבה הוא נמצא. לאחר מכן הנתבן נותן לעצמו Interface ID ייחודי (EUI-64). הנתב שולח כל 200 שניות RA ב- Multicast לכתובת 1::FF02 שמייצגת את כל התקנים ברשת.



Redirect (Re)

הנתב מודיע ללקוח שיש נתב ברירת מחדל טוב ממנו.

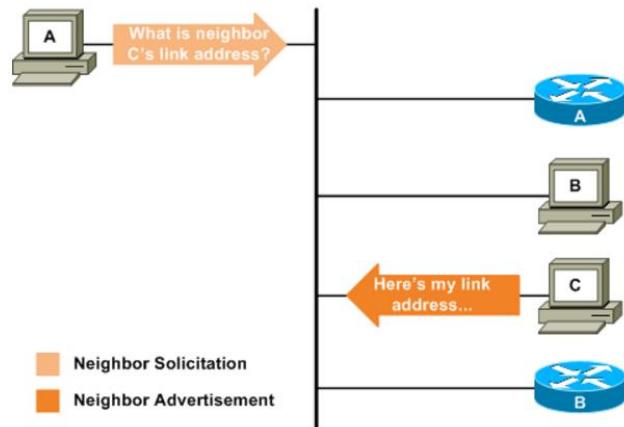


Neighbor Solicitation (NS)

תחליף ל-ARP. הלקוח מhapus כתובת פיסית.

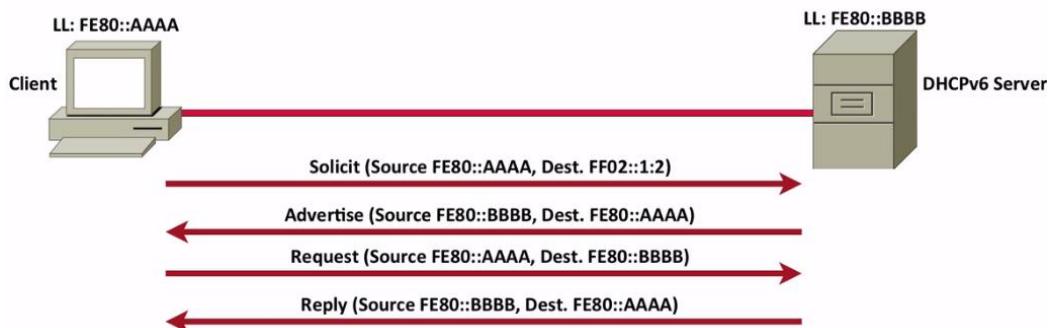
Neighbor Advertisement (NA)

תגובה לבקשת.

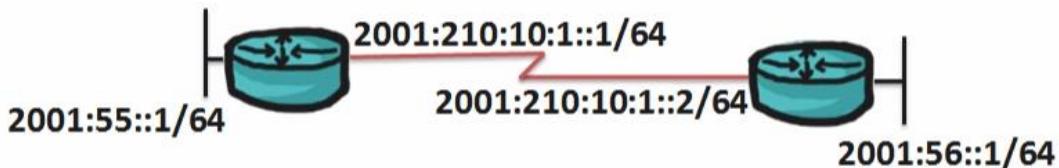


DHCPv6

כאשר מחשב נותן לעצמו כתובת, הוא לא ידוע מה כתובתה של שירות DNS ופרטים נוספים ששרת DHCP מספק ולכן עדין יש צורך בשירות DHCPv6. IPv6 אין שידור Broadcast ולכן, תחילה הלכה משדר RS ובתגובה יקבל RA עם הפרטים של השירות DHCP. לאחר מכן, הלכה פונה לשרת DHCPv6 ב- Multicast לשורה עליה מזינים כל הרשמי DHCP (FF02::1:2). מחשב יכול לחת לעצמו כתובת ושרת DHCPv6 ישלים ללקוח את הפרטים החסרים.



הגדרת כתובת IPv6 לנット



הפעלת IPv6

כברירת מהדיל IPv6 לאאפשר בנתב.
כדי לאפשר זאת נשתמש בפקודה:

Router(config)#**ipv6 unicast-routing**

כאשר אנו נותנים לממשק כתובת IPv6, אז הממשק מקבל גם בצורה אוטומטית כתובת מסוג-link-local. אם לא נתנו לממשק כתובת ואנו רוצים שהממשק יקבל כתובת link-local addresses אז נשתמש בפקודה זו.

Router(config-if)#**ipv6 enable**

הגדרת כתובת לממשק

Router(config-if)#**ipv6 address {2001:210:10:1::1}/{64}**

שימוש בשיטת eui-64:

Router(config-if)#**ipv6 address {2001:210:10:1}::/{64} eui-64**

הגדרת ידנית של Link-Local Address - היתרון במתן כתובת ידנית זה מתן כתובת פשוטה הכתובת משמשת לזיהויו הנtent באותו מקטע בלבד ולtent יכולם להיות כמה משים עם כתובות זהה.

Router(config-if)#**ipv6 address {fe80::1} link-local**

כדי לבדוק שהנתב קיבל את הכתובת, נשתמש בפקודה:

Router# **show ipv6 interface brief**

```
R0#show ipv6 interface brief
FastEthernet0/0           [administratively down/down]
FastEthernet0/1           [administratively down/down]
Serial0/0/0               [down/down]
    FE80::20B:BEFF:FE24:3401
    2001:210:10:1::1
```

הגדרת ניתוב סטטי בסביבת IPv6

R0(config)#**ipv6 route {2001:56::/64} {2001:210:10:1::2}**

בדיקות תקינות טבלת הניתוב

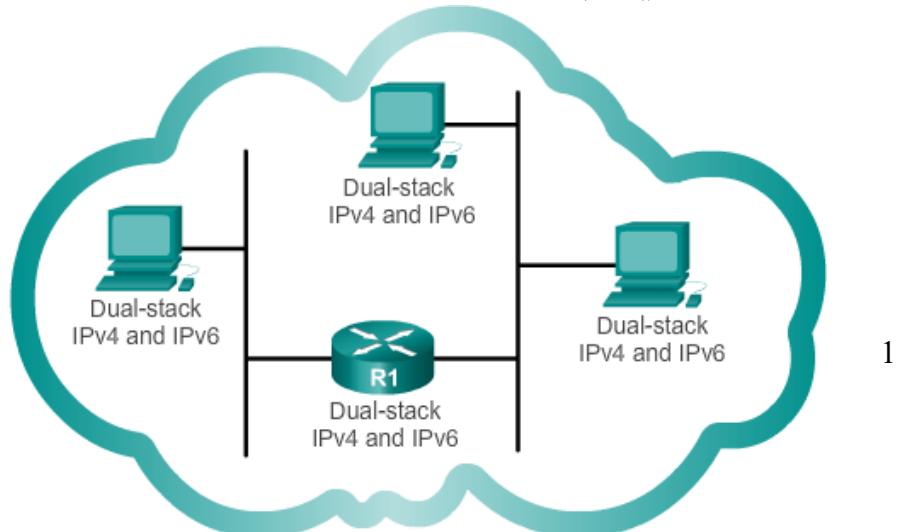
Router# **show ipv6 route**

מעבר כתובות מי IPv4 ל-IPv6

IPv6 Internet Engineering Task Force (IETF) פיתח שלוש שיטות למעבר מי IPv4 ל-

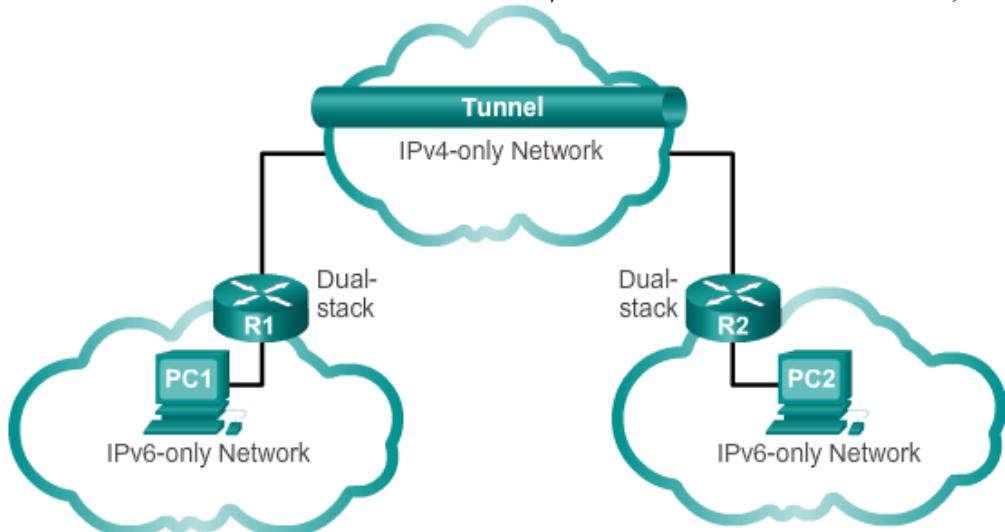
Dual Stack

זו השיטה הקלה ביותר. השיטה מאפשרת תמיכת IPv6 ו-IPv4 בכרטיס רשת. כך פעולים על הכרטיס רשת שני מחסנויות פרוטוקול של TCP/IP ולכרטיס רשת יש שני כתובות IP. הבעיה היא, לא כל התקן תומך ב-IPv6.



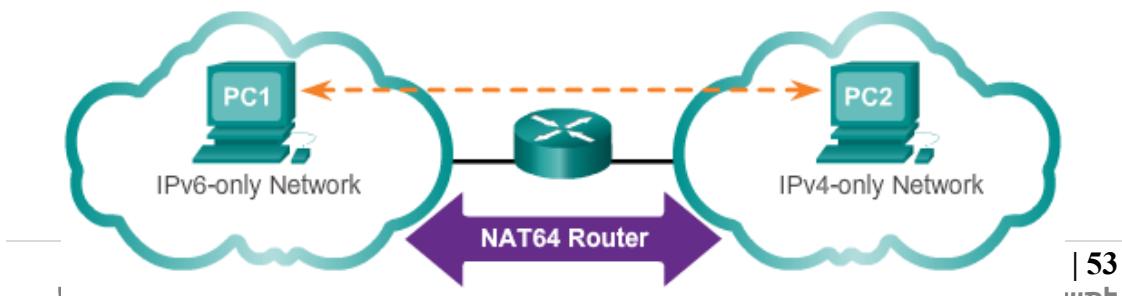
Tunneling

בשיטה זו, מסוג IPv6 נארז מחדש לתוך IPv4 Packet.

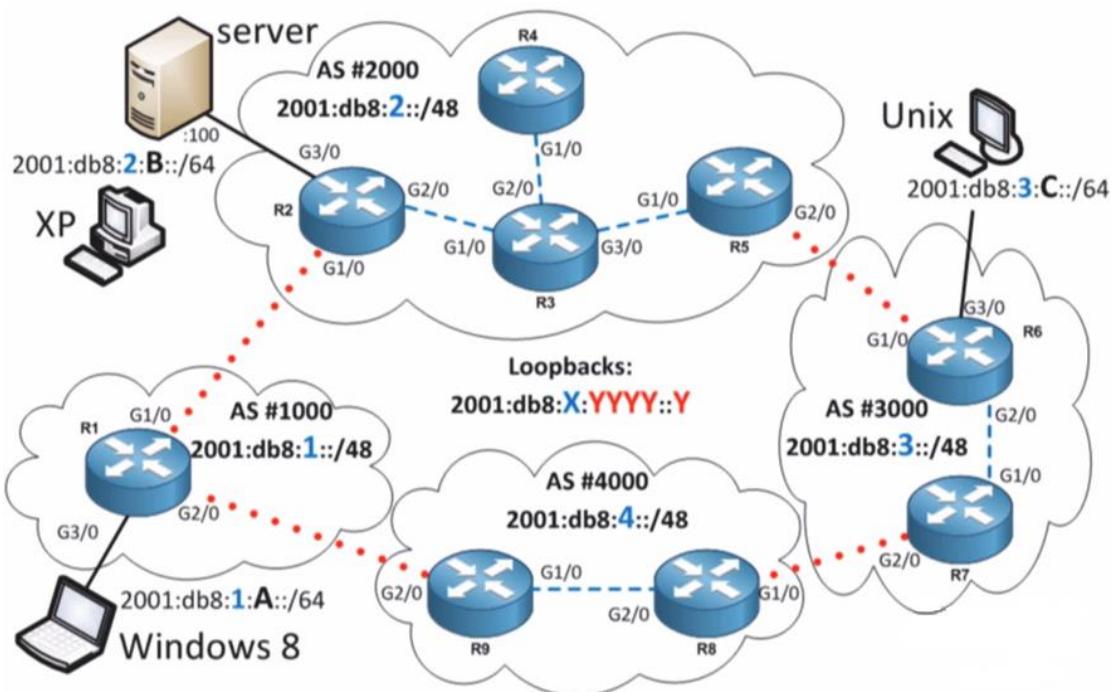


Translation

בשיטה זו, טכנולוגיית Network Address Translation 64 (NAT64) מאפשרת תרגום כתובת IPv6 לכתובת IPv4 וההפך.

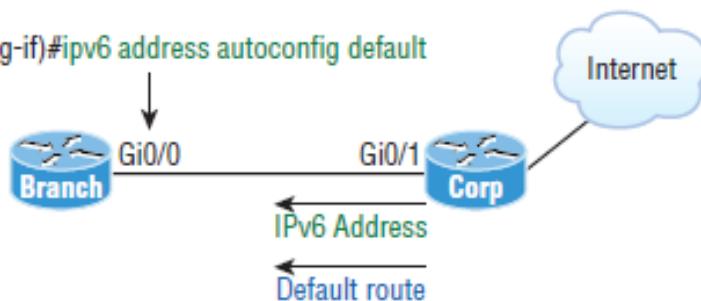


IPv6 Subnetting



הגדרת נתב לקבלה כתובת בצורה אוטומטית:

```
Branch(config-if)#ipv6 address autoconfig default
```



Cisco Discovery Protocol (CDP)

זה פרוטוקול שפותחה cisco ופועל בהתקנים של .cisco.
CDP מאפשר למנהל הרשות לאסוף מידע על התקנים שהוברים לרשת.
הפרוטוקול פועל בשכבה שנייה ולפניהם גם במשקים ללא כוותת IP.
כאשר CDP פועל על התקן, CDP שולח ומתקבל מידע רק מהתקנים שכנים (neighbors).

```
Router#show cdp
Global CDP information:
  Sending CDP packets every 60 seconds
  Sending a holdtime value of 180 seconds
  Sending CDPv2 advertisements is enabled
```

קובע את קצב שליחת cdp packets דרך המשകים הפעילים:
(Packet Tracer) (לא ניתן עלי ידי)

```
Router(config)#cdp timer {5-254}
```

קובע לכמה זמן cdp packet בתוקף:
(Packet Tracer) (לא ניתן עלי ידי)

```
Router(config)#cdp holdtime {10-255}
```

הפעלה או סגירת CDP

```
Router(config)#cdp run
```

```
Router(config)#no cdp run
```

הפעלה או סגירת CDP, על משק מסוים:

```
Router(config-if)#cdp enable
```

```
Router(config-if)#no cdp enable
```

מראה פרטיים כללים על השכנים:

```
Router#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
Device ID    Local Intrfce   Holdtme    Capability      Platform  Port ID
Switch       Fas 0/0        120          S              2950      Fas 0/1
Switch       Fas 0/1        120          S              2950      Fas 0/1
Router       Ser 0/0/1      120          R              C1841     Ser 0/0/0
```

מראה פרטיים מפורטים על השכנים:

```
Router#show cdp neighbors detail

Device ID: Switch
Entry address(es):
Platform: cisco 2950, Capabilities: Switch
Interface: FastEthernet0/1, Port ID (outgoing port): FastEthernet0/1
Holdtime: 174

Version :
Cisco Internetwork Operating System Software
IOS (tm) C2950 Software (C2950-I6Q4L2-M), Version 12.1(22)EA4, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2005 by cisco Systems, Inc.
Compiled Wed 18-May-05 22:31 by jharirba

advertisement version: 2
Duplex: full
-----
Device ID: Router
Entry address(es):
  IP address : 192.168.1.1
Platform: cisco C1841, Capabilities: Router
Interface: Serial0/0/1, Port ID (outgoing port): Serial0/0/0
Holdtime: 174

Version :
Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M), Version 12.4(15)T1, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Wed 18-Jul-07 04:52 by pt_team

advertisement version: 2
Duplex: full
```

Link Layer Discovery Protocol (LLDP)

זה פרוטוקול לשימוש פתוח (open standard) שעבוד בקרה דומה ל- CDP.
נתמך על ידי ציוד של כל חברה כולל חברות של Linux.
IEEE 802.1AB

הפעלה או סגירת LLDP

Router(config)#**lldp run**

Router(config)#**no lldp run**

הפעלה או סגירת LLDP, על מנת מסוים:

Router(config-if)#**lldp transmit**
Router(config-if)#**lldp receive**

Router(config-if)#**no lldp transmit**
Router(config-if)#**no lldp receive**

שאר הפקודות דומות מאוד לפקודות של CDP.

Network Time Protocol (NTP)

אחד הסיבות המרכזיות שהשעונים של כל ההתקנים ברשת צריכים להיות מסונכרנים ומעודכנים זה יכולת לעקוב אחר logs בסדר הנכון. בנוסף, SSH ו-VPN משתמשים בתעוזות דיגיטליות ואם השעון לא מעודכן יכול להיות שהתעוזות יהיו לא בתוקף ולכך השירותים האלה יכשלו.

כדי לראות את השעון השתמש בפקודה **show clock**

ניתן לעדכן את השעון בשני דרכים:

- **ידנית** - בעזרת הפקודה {day month year}router# **clock set 14:12:00 10 feb 2016**לדוגמא:
▪ **אוטומטית** – שימוש ב- (NTP) Network Time Protocol

בסביבה מואובטחת, לא נרצה שככל ההתקנים ברשת יקבלו עדכון באמצעות NTP Server שנמצא באינטרנט כי זה חושף אותם לסלנות. לכן, בדרך כלל יהיה ברשות שרת NTP שמתעדכן אוטומטית דרך האינטרנט וכל ההתקנים ברשות מתעדכנים דרכו.

NTP Client יכול לעבוד בשלוש דרכים:

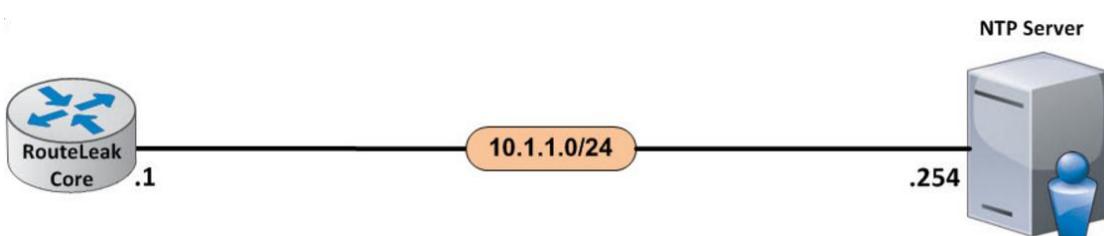
- לחשאל את השירות.
- להאזין להודעות NTP Multicast
- להאזין להודעות NTP Broadcast

הגדרת NTP Client

1. כדי למצוא כתובת של NTP Server נתרגם את השם Pool.ntp.org לכתובת IP. לאחר מכן נקבע שם נמצאים מספר שרת NTP.

2. הגדרת Timezone {2} Router(config)#clock timezone {israel} {2} נitin לבדוק שההגדרה הצליחה באמצעות הפקודה show clock

3. הגדרת הכתובת של NTP Server R1(config)#ntp server {ip address} נitin לבדוק שההגדרה הצליחה באמצעות הפקודה show ntp associations



```
R1(config)#ntp master
```

System Message Logging (Syslog)

להתקנים ברשת קיימים מנגנון שמאפשר להם להודיעו כאשר מתרחשים אירועים מסוימים וזאת באמצעות שליחת הודעות מערכת. קריית הודעות מערכת מאפשרת לבדוק את מצב התקן ועזרה באיתור תקלות. תפקיד פרוטוקול syslog לשЛОוח הודעות מערכת.

כברירת מחדל, נתבים ומוגדים של Cisco שולחים את כל ההודעות ל- console ובחלק מגרסאות IOS, המכשיר גם אוגר הודעות יומן ב- .buffer.

עדים עבור הודעות syslog כוללים:

- Logging buffer (זיכרון RAM בתוך התקן) - פעיל כברירת מחדל
- Console line
- Terminal line
- Syslog server

כדי לראות את ההגדרות של syslog ואת ה הודעות ששמורות ב- buffer, השתמש בפקודה:

```
R1# show logging
```

כדי לאפשר שליחת ה הודעות ל- buffered/console, השתמש בפקודות:

```
R1(config)# logging console
R1(config)# logging buffered
```

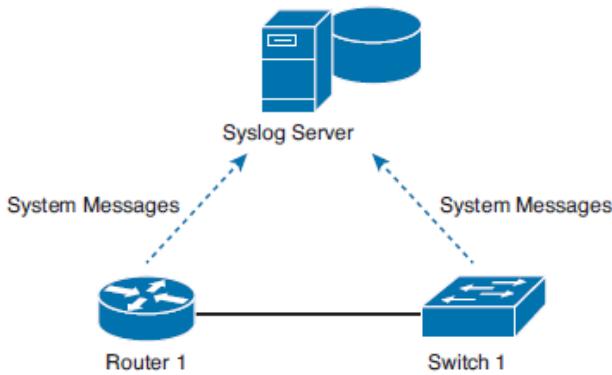
Severity Level

אחד המרכיבים החשובים בהודעת מערכת זה חומרת ההודעה. Severity Level מאפשר לנו לשלוט בסוגי ה הודעות נשלחות ל- Syslog server. ככל שהמספר נמוך יותר, כך ההודעה קריטית יותר. רמות 0-4 מייצגים אירועים שעולים להשפיע על התקן ברכיניות, ואילו רמות 5-7 מייצגים אירועים פחות חשובים.

Level	Level Name	Explanation
0	Emergency	The system may be unusable.
1	Alert	Immediate action may be required.
2	Critical	A critical event took place.
3	Error	The router experienced an error.
4	Warning	A condition might warrant attention.
5	Notification	A normal but significant condition occurred.
6	Informational	A normal event occurred.
7	Debugging	The output is a result of a debug command.

כדי להגדיר איזה הודעות תישלח (Severity Level).
 לדוגמה, כדי להגביל את ההודעות לרמות 4-0, השתמש באחד משתי הפקודות:
R1(config)# logging trap {4}
R1(config)# logging trap {warning}

אפשר גישה להודעות מערכת ללא צורך בגישה פיסית להתקן.
 זו השיטה הנוחה והנפוצה ביותר לגישה להודעות מערכת.



כדי לשולח הודעות ל- syslog server, יש להגדיר את הכתובת של syslog server:
R1(config)# logging host {IP address of the syslog server}

Time-stamp
 כבירות מחדל, חותמת הזמן שמופיע בהודעה היא משך הזמן שהתקן פועל.
 עיל יותר שבהודעות תופיע חותמת זמן וכן השעון בהתקן צריך להיות מעודכן.

R1(config)#service timestamps log datetime msec

ניתן לקבוע שבקום חותמת הזמן, הרשומות יהיו ממוספרות.

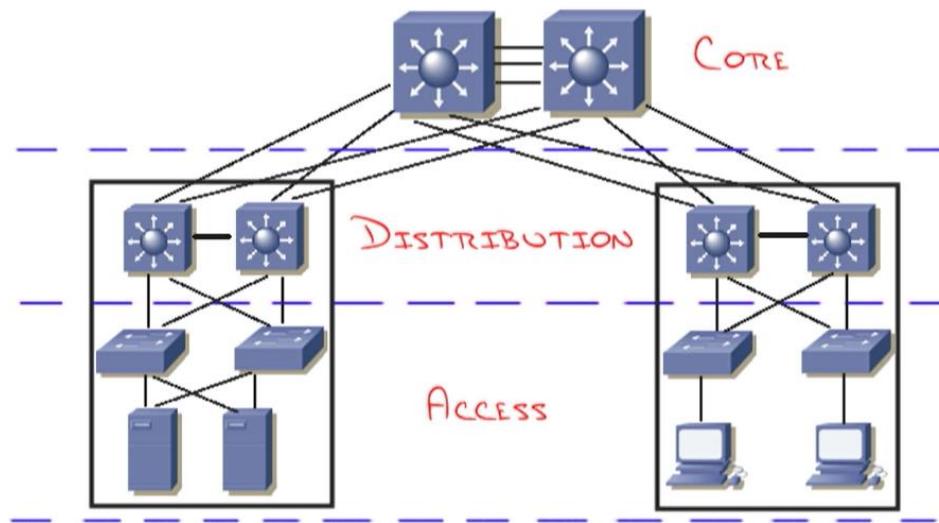
R1(config)#no service timestamps
R1(config)#service sequence-numbers

קיימות כמה גרסאות של תוכנות חופשיות ושיתופיות שמאפשרות הקמת שרת syslog
 לדוגמה: .kiwi syslog server
 Splunk Light

Spanning Tree Protocol

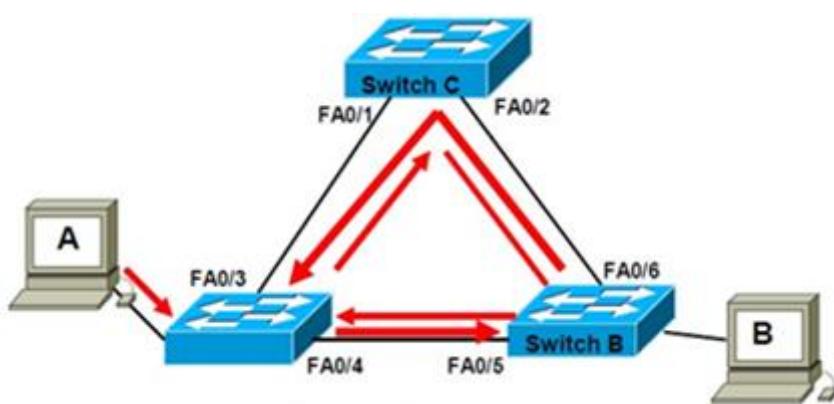
(יתירות) Redundancy

כישלון של ציוד, קישור או Port קריטי, יכול לגרום להשבחת חלקים נרחבים ברשת. כדי לפתרו בעיה זו, ניתן ליצור מספר מסלולים שונים אל יעד מסוים על ידי התקנת ציוד או קישור כפול באזוריים קריטיים וכן לשמר על רמת אמינות גבוהה. Redundancy מלא יקר ולכן יש צורך לאזן בין הצורך ב贅ית לבין עלות ייצור.



Broadcast storm

הסכנה המרכזית בייצור Redundancy בציוד של שכבה 2, זה Broadcast storm. Broadcast storm מטוג Frame מסוג Broadcast, המתג משדר אותו דרך כל הממשקים מלבד המשק דרכו - Frame נכוון. Frame אין מנגנון כמו TTL (TTL), ולכן יושודר בין המתגים בצורה מעגלית ללא סוף וכך ינצל את כל רוחב הפס ויגרום להשבחת הרשת.



כיצד (Spanning Tree Protocol (STP) מתר לולאות?

STP זהו המנגנון שמודע לאטר ולבטל לולאות מיתוג ברשת. כדי לגלוות האם קיימות ברשת לולאות מיתוג, המתג משדר כל 2 שניות BPDU Packet (Bridge Protocol Data Units). אם ה- BPDU Packet חוזר למtag דרך יציאה אחרת, אז המשמעות היא שקיימת ברשת לולאה. תפקידו של STP לסגור את הלולאה. תהליך הלימוד של STP נקרא התכנסות (converged) ובסיום התהליך אין לולאות ברשת. גבולותיו של STP זהה ה- Broadcast Domain (שזה גם VLAN) ולכן יש צורך בניהול STP בצורה נפרדת בכל Broadcast Domain.

מה תפקידו של ה- Root Bridge ?

כדי שהרשת תעבור בצורה ייעילה, חשוב שהמתג שנבחר לשמש כ- Root Bridge יהיה במרכז הרשת. זהו המתג שמשמש כנקודות יחוות למתקנים אחרים ברשת. כאשר מתג מסוים מתר לולאה, עליו לבחור איזה דרך צריכה להישאר פתוחה ואיזה דרך אפשר לסגור וזאת במטרה לסגור את הלולאה. המתג עושה זאת על-ידי מציאת הנתיב הטוב ביותר ל- Root Bridge וחסימת כל נתיב נוסף.

כיצד מתג בוחר את הדרכ הטובה ביותר ל- Root Bridge

זהו מספר שמקבל כל ממשק בתבנית Cost משקף את מהירות החיבור מי המשק במתג ועד ל- Root Bridge. ככל שהמספר נמוך יותר, כך הדרך ל- Root Bridge מהירה יותר.

בחירת המשק אותו נרצה לסגור מתחבע לפי העדיפויות הבאה:

1. המשק עם ה- cost הגבוה ביותר ייסגר.
2. אם ה- Cost בשני ממשקים שווה, אז נבדק את ה- (BID) Bridge ID הגבוה ביותר ייסגר המשק.
3. המספר ממשק הגבוה ביותר ייסגר.

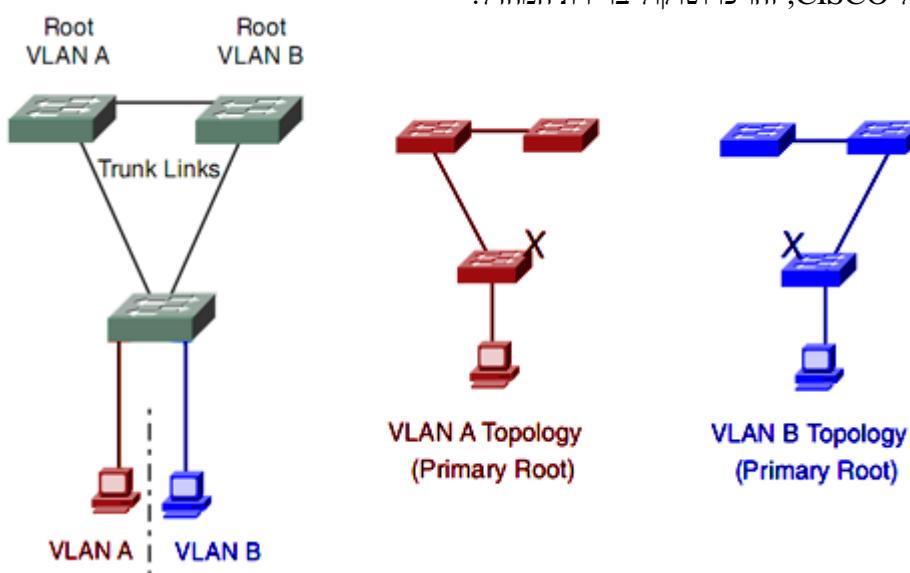
Link Bandwidth	Cost
10Mbps	100
100Mbps	19
1Gbps	4
10Gbps	2

סטנדרטים של (STP) Spanning Tree Protocol

בשנת 1993 יצא הסטנדרט הראשון של STP. ה프וטוקול עובד בשכבה השנייה של מודל OSI על פי תקן **802.1D**. הביעה המרכזית עם STP היא שולחן 30-50 שניות עד שימוש נכון למצב פעיל.

Per-VLAN Spanning Tree (PVST+)

זהו פרוטוקול שפיתחה CISCO. בשיטה זו, לכל VLAN יש Root Bridge נפרד ולכל ניתן לנצל את רוחב הפס בצורה טוביה יותר. במקרה של Loop, אין צורך לסגור חיבור לגמרי, אפשרי להשאירו פעיל ב- VLAN מסוים בלבד וכך ניצול רוחב הפס טוב יותר. במקרים של CISCO, זהו פרוטוקול ברירת המחדל.



Rapid STP (RSTP)

הפרוטוקול עובד בשכבה שנייה של מודל OSI על פי תקן **802.1W**. RSTP מעביר ממושך מצב blocking במצב 2 forwarding (STP מצבually 50 שניות). מה שגורם להבדל בין STP ו-RSTP הוא ש-RSTP פסיבי (מחכה כל פעם לבדוק מה מצב המשק) ו- RSTP אקטיבי.

Per-VLAN Rapid STP (PVRST)

זהו פרוטוקול שפיתחה CISCO. Rapid STP (RSTP) עם Per-VLAN Spanning Tree (PVST+).

MULTIPLE Spanning Tree Protocol (MSTP)

זהו הסטנדרט שפותח כתחליף תואם ל- Per-VLAN. MSTP (PVRST) הפעוטוקול עובד בשכבה שנייה של מודל OSI על פי תקן **802.1S**.

כיצד נבחר ה- Root Bridge?
Bridge ID (BID) הוא מספר ייחודי שיש לכל מtag.
 המtag עם ערך BID הנמוך ביותר, יבחר כ- Root Bridge.

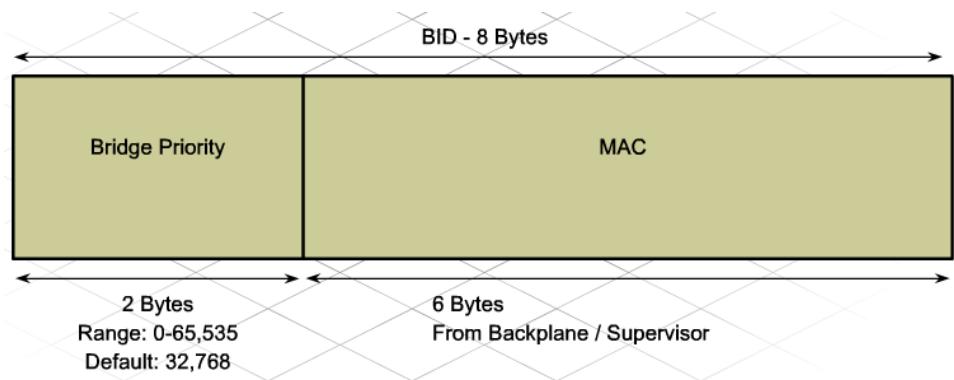
BID מורכב מי שני חלקים:

- **Bridge Priority** (0-65535), הערכאים יכולים להשתנות בקפיצות של 4096.
- **Custody MAC**.

כברירת מחדל, הערך של Bridge Priority הוא 32,768 וילך, כברירת מחדל המtag עם הכתובת MAC הנמוכה ביותר נבחר לשמש כ- Root Bridge.

SAMPLE BRIDGE IDs:

32768.00A0.1101.B011
 32768.00A0.FF01.6689
 32768.0010.FF32.991B



נשתמש בפקודה **show spanning-tree** כדי לראות את המידע הבא:

- מי ה- (Root ID) Root Bridge.
- מה ה- (Bridge ID) bridge priority.
- עדיפות הממשקים (Cost).
- מצב המשקיפים.

Switch#show spanning-tree

```

Switch#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID      Priority    4097
                Address     00E0.F74E.3CC9
                Cost        38
                Port       1(FastEthernet0/1)
                Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

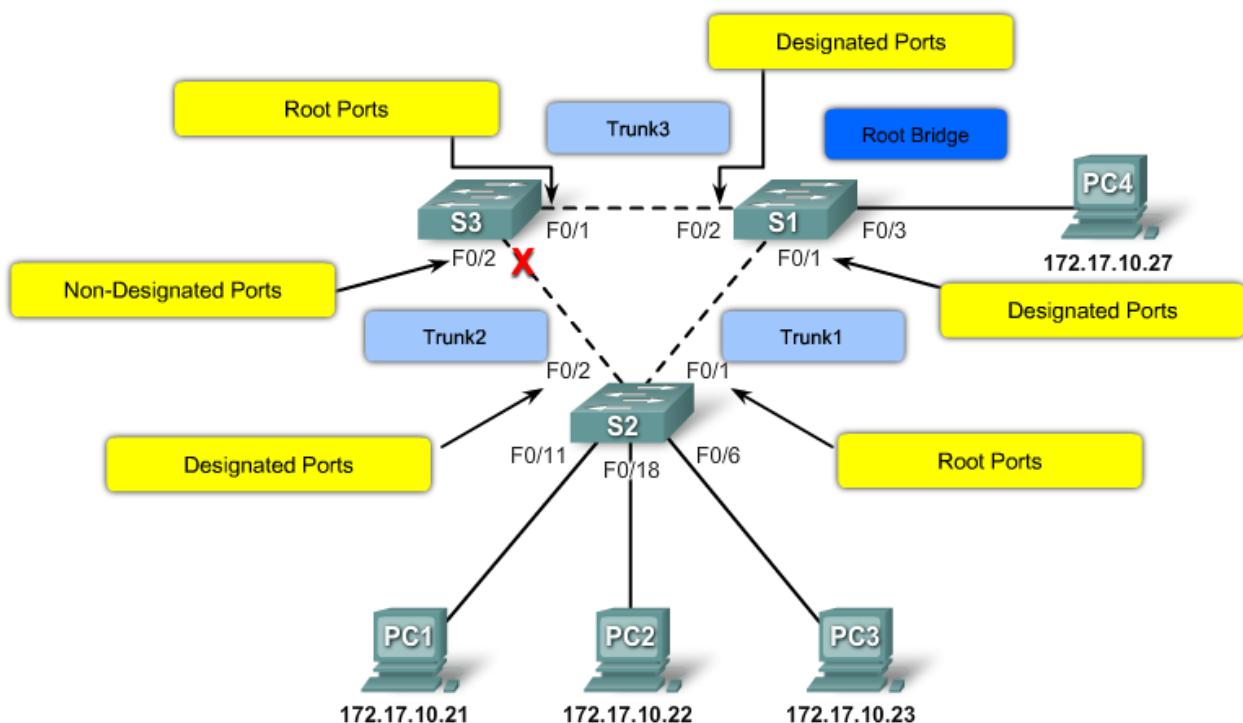
  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
                Address     0001.C7D7.A038
                Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
                Aging Time 20

  Interface      Role Sts Cost      Prio.Nbr Type
  ----- -----
  Fa0/1          Root FWD 19      128.1    P2p
  Fa0/2          Altn BLK 19      128.2    P2p

```

תפקידי PORTS

- Root Port** • לכל מtag יש Root Port אחד, שהוא הנתיב המהיר ביותר לRoot Bridge.
- Designated Port** • זהו Port שמעביר מידע. בחיבור בין שני מטגיים, צד אחד חייב להיות Designated Port.
- זהו Port שמסוגל מידע. Non-Designated Port יכול להיות Root Port או Designated Port.
- Non-designated port** • זהו Port עם ה-cost הגבוה ביותר ב-loop ולכן זה port חסום (מעביר BPDUs).
- Disabled Port** • זהו port לא פעיל (administratively shut down).

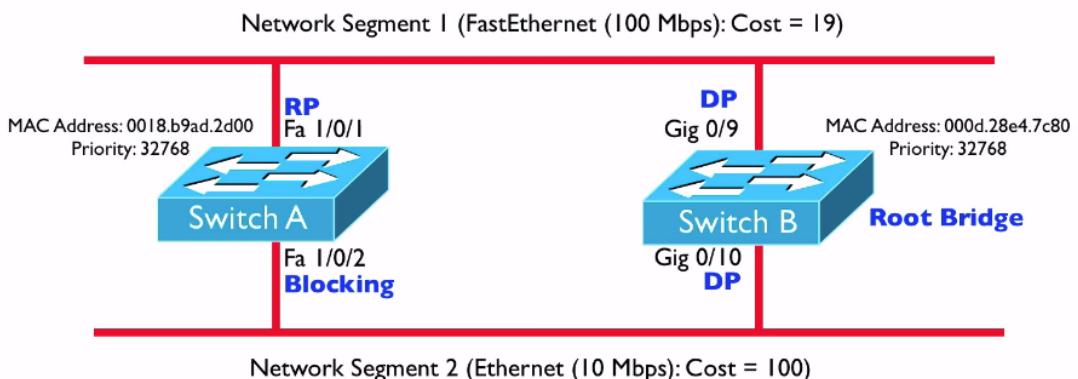


סוגי RSTP Ports

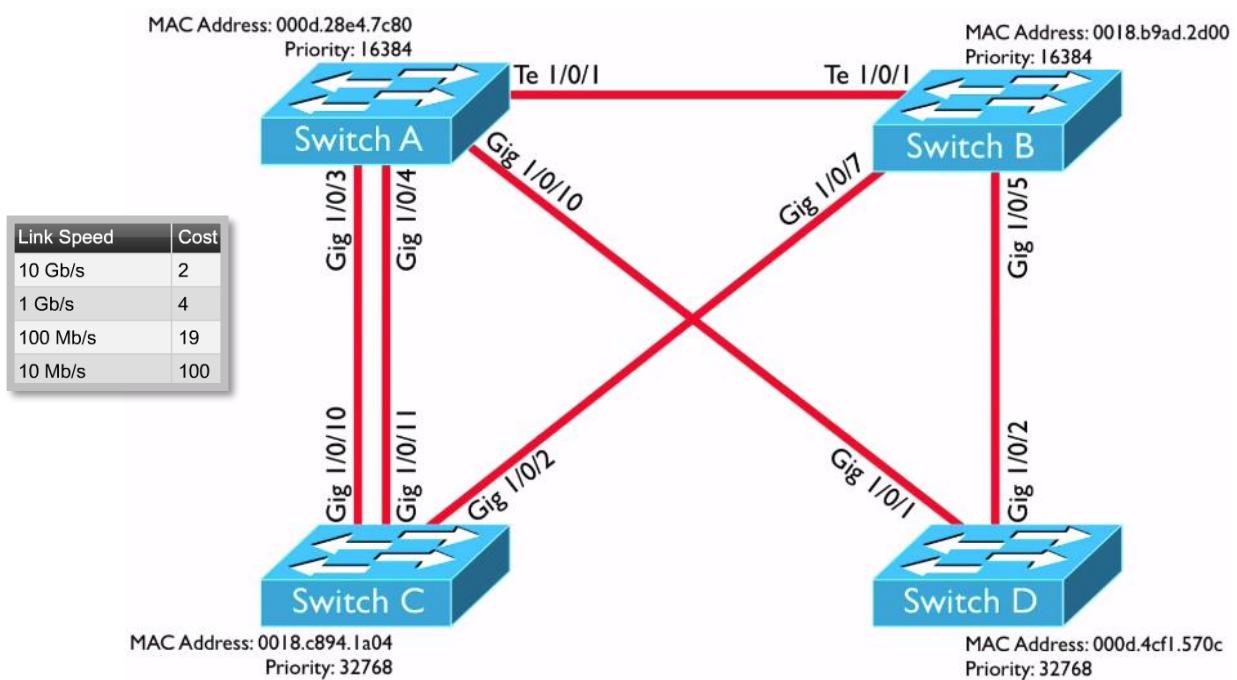
- .Root Bridge - Root Port • הזרק המהירה ביותר לRoot Bridge.
- .Designated Port • מעביר מידע.
- .Alternate Port – חסם מעבר מידע כדי למנוע Loop.

תהליך בחירת Port חסום

1. תחיליה נבחר ה- Root Bridge . בכל Broadcast Domain יש רק אחד Root Bridge.
2. לאחר מכן, כל מתג בוחר איזה Port ישמש כ- Root Port .
3. בין שני מתגים שכנים, Port אחד חייב להיות במצב של Designated Port .
4. מה שנשאר נחסם.



תרגיל



מצבי PORTS

- כasher אנו מחברים התקן למtag, lokh בין 30 ל- 50 שניות עד שהוא מעביר מידע לרשת.
אלו השלבים ש- Port עובר כאשר הוא תומך ב- STP:
- (20 שניות) - המtag מקשיב לתעבורת BPDU ולומד את מצב הרשות.
 - (15 שניות) – המtag שולח/מקבל הודעות BPDU.
 - (15 שניות) – המtag לומד Mac Address של התקנים שכשרים ל- Port.
 - – ה- Port במצב פעיל ומעביר מידע.

ב- RSTP קיימים רק שלושה מצבים port:

- локח 2 שניות לעברו ממצב learning למצב forwarding •
learning •
forwarding •
discarding (חסימה) •

PortFast

כיום המחשבים מהירים מאד ולכן יתכן שמערכת הפעלה במחשב תעלת בפחות מי 30 שניות, ולכן יש צורך בזמינות ממشك בפחות מי 30 שניות.
forwarding PortFast מנטREL את STP ולכן עובר מיד ממצב blocking למצב •
נתממש בהגדירה זו רק ל- Ports שכשרים אליהם מחשבים.

```
Switch(config-if)#spanning-tree portfast
```

BPDU Guard

את הגדרה זו כדי להפעיל במידה והשתמשנו ב- PortFast.
הגדרה זו, גורמת לכך שאם יגיע ממشك BPDU Packet יכנס למצב ErrDisable ולא יעביר יותר מידע. במצב זה, רק ניהול רשות יכול לפתח חזרה את הממשק.

```
Switch(config-if)#spanning-tree bpduguard enable
```

כדי לשנות עדיפות של מתג:

```
Switch(config)#spanning-tree vlan {1} priority {4096}  
Switch(config)#spanning-tree vlan {10,20,30,40} root {primary}
```

ה חוזרת העדיפות לבירית מחדל:

```
Switch(config)#no spanning-tree vlan {1} Priority
```

כדי לשנות את ה- Cost לממשק:
S2 (config)#interface f0/1
S2 (config-if) #spanning-tree cost 25

כדי לחזור לערך בירית מחדל:
S2 (config)#interface f0/1
S2 (config-if) #no spanning-tree cost

הגדרת Rapid STP (RSTP)

```
Switch(config)#spanning-tree mode rapid-pvst
```

שנム מספַר טִימָרִים ב-**STP**

Hello - זמן Hello הוא הזמן בין משלוח למשЛОח שנשלח דרך היציאה שווה ל 2 שניות כבירת מחדר, אבל אתה יכול לכוון את הזמן להיות בין 4 ל-30 שניות.

Forward delay - זמן Forward Delay הוא שווה ל-15 שניות כבירת מחדר, אבל אתה יכול לכוון את הזמן להיות בין 1 ל 10 שניות.

Max Age - טימר הגיל המקסימלי שולט באורך הזמן המקסימלי שעובר לפני שהמתק שומר את פרטי תצורת ה- BPDU שלו. הזמן הוא 20 שניות כבירת מחדר, אבל אתה יכול לכוון את הזמן להיות בין 6 ל 40 שניות.

Open Shortest Path First (OSPF)

- .Link State Protocol ניתוב לשימוש פתוח (open standard) שעובד בשיטת ברשנות. כיוון זהה הפרטוקול ניתוב הנפוץ. Administrative distance של OSPF הוא 110. עדיפות הנתיב (cost) נקבעת לפי רוחב הפס של כל הממשקים מרנתב עד לעד. OSPFv2 תומך ב-IPv4 ו-OSPFv3 תומך ב-IPv6.

OSPF Area

כאשר הרשת גדולה בסיס נתוניים של הנטב גודל וזה גורם למספר חסרוןנות:

- העומס על המעבד והזיכרון של הנטב גדול כי צריך לעבד מידע רב.
- כאשר מתרחש שינוי בטופולוגיה הרשת, תהליך ההתקנסות (convergence) איטי.
- שינוי מצב של ממשק (up/down/קע) בנטב, מאלץ את כל הנטבים ברשת לבצע חישובים מחדש.

הפתרון, חלוקת טופולוגיה הרשת למספר אזוריים. Cisco ממליצה עד 50 נתבים באזור אחד. כך כל הנתבים באותו Area מכילים טבלת טופולוגיה זהה וכל נתב מודע רק לאזור אליו הוא שייך.

תרונות עובדה ברשת היררכית (OSPF Areas):

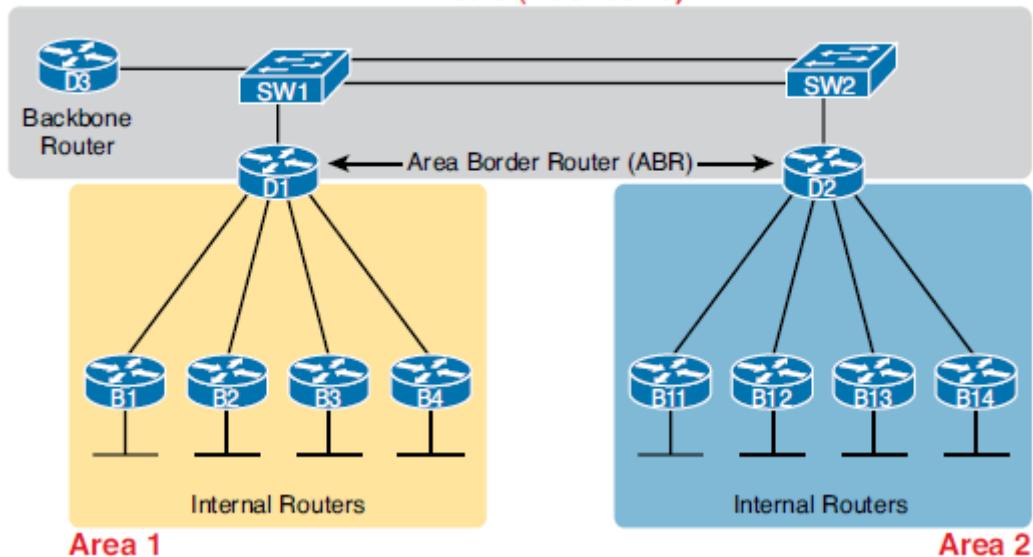
- פחות עדכנים עוברים ברשת.
- יש פחות עומס על המעבד והזיכרון של הנטב בכלל שיש פחות חישובים לבצע.
- תהליך ההתקנסות מהיר יותר.
- תקלת באזור אחד לא תשפיע על שאר האזוריים.
- טבלאות הניתוב קטנות יותר.

כל האזוריים חייבים להתחבר ל Area 0 שמשמש כ Backbone.

Area Border Router (ABR)

נתבים שמחברים בין Areas נקראים Area Border Router (ABR) והם מכירים יותר מאשר אחד. נתב ששימש כ ABR יודע לבצע CIDR (Summarization) של אזור ולכן חשוב לתכנן Area בצורה היררכית כדי שנitin יהיה לבצע צמצום כתובות רשת בצורה יעילה.

Area 0 (Backbone)



(RID) Router ID

בסביבת OSPF, לכל נתב מוגדר RID שמצויה אותו כיחודי.

תהליך בחירת ה- Router ID

- הגדירה ידנית. Router(config-router)#router-id {router-id}

- אם לא מוגדר ידני, הממשק loopback עם הכתובת IP **הגבולה** ביוור, תשמש כ-

```
R1(config)# interface loopback 0
R1(config-if)# ip address 1.1.1.1 255.255.255.255
```

- אם לא הוגדר ידנית ואין ממשק loopback, אז הממשק הפעיל עם הכתובת **הגבולה** ביוור נבחר כ- Router ID.

כדי לבדוק מהו ה- Router id השתמש בפקודה - show ip ospf

אם נעשה שינוי ב- Router ID לאחר ש- OSPF התחיל לעבוד, יש להפעיל מחדש את OSPF. הפקודה לא נטמכת על-ידי Packet Tracer.

```
Router(config)#clear ip ospf process
```

טבלת השכנים (Neighbor table)

הטבלה מכילה את השכנים שהנתב מכיר.

```
R3# show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
1.1.1.1	0	FULL/-	00:00:33	10.1.13.1	Serial0/0/0
10.1.24.4	1	FULL/DR	00:00:35	10.1.3.130	GigabitEthernet0/0.342
10.1.24.4	1	FULL/DR	00:00:36	10.1.3.4	GigabitEthernet0/0.341

תהליך הקמת קשרי שכנות

כדי לאגלו נתבים שכנים ולתזקק את הקשר איתם, הנתב שולח Hello packets דרך כל הממשקים שעובדים ב- OSPF. התהליך כולל מספר שלבים:

- דרך ממשק במצב down, לא ניתן לקיים יחס שכנות וلن.

תהליך הקמת קשר עם השכנים מתחילה כאשר הממשק עובר במצב up link.

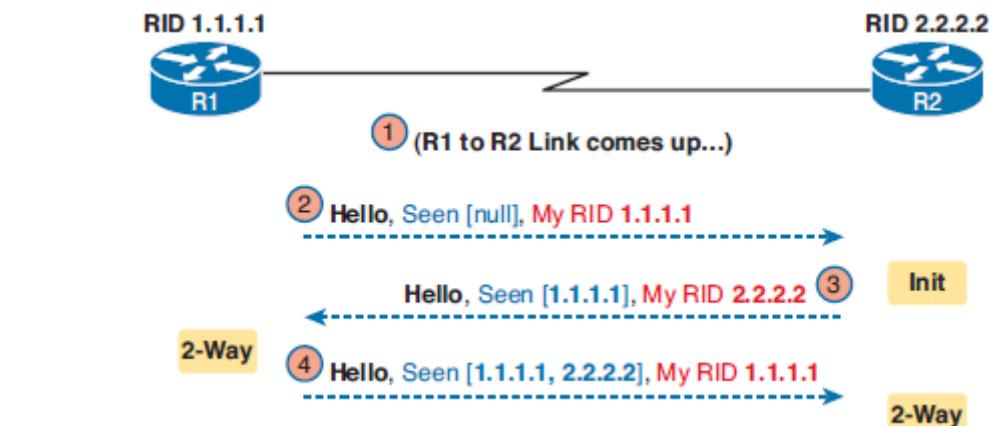
- R1 שולח Hello packet ב- Multicast לכתובת 224.0.0.5.

R2 לומד על קיום R1 ורושם אותו כשכן זמני (Init).

R2 שולח Hello packet בוא הוא מציין את ה- RID של R1.

כאשר R1 מקבל את החבילה הוא מכניס את R2 לטבלת השכנים (2-Way).

- R1 שולח Hello packet ל- R2 וכותזאה מכך R2 עבר למצב 2-Way.

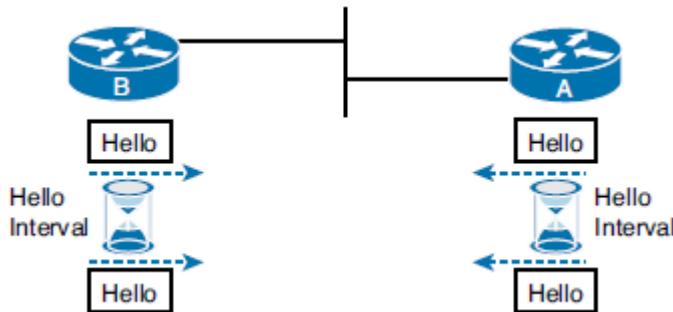


תהליך תחזוקת הקשר עם השכנים

בהתהליך זה קיימים שני פרמטרים חשובים:

- **Hello Interval**

כברירת מחדל מוגדר לשלוח כל 10 שניות ובממשק סריאלי כל 30 שניות.

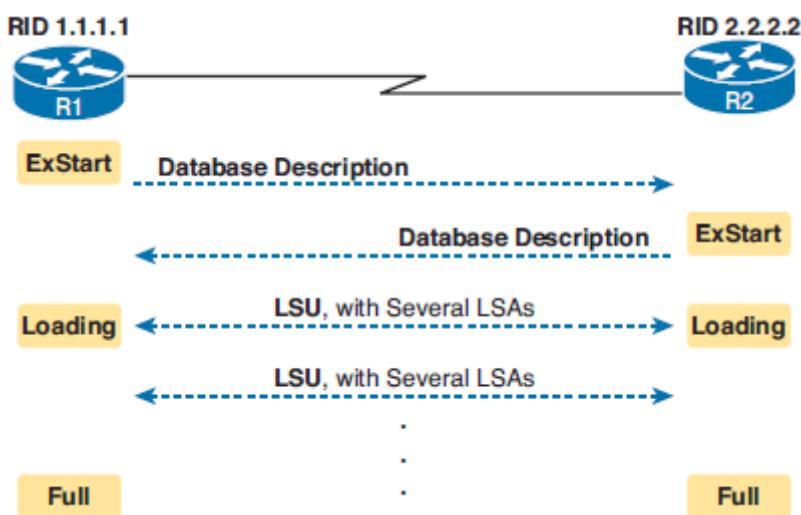


- אם נtab לא קיבל hello packet במשך 40 שניות, אז השכן נחשב כ-לא זמין. במשק סריאלי לאחר 120 שניות השכן לא זמין.

תהליך הסנכרון בין שכנים

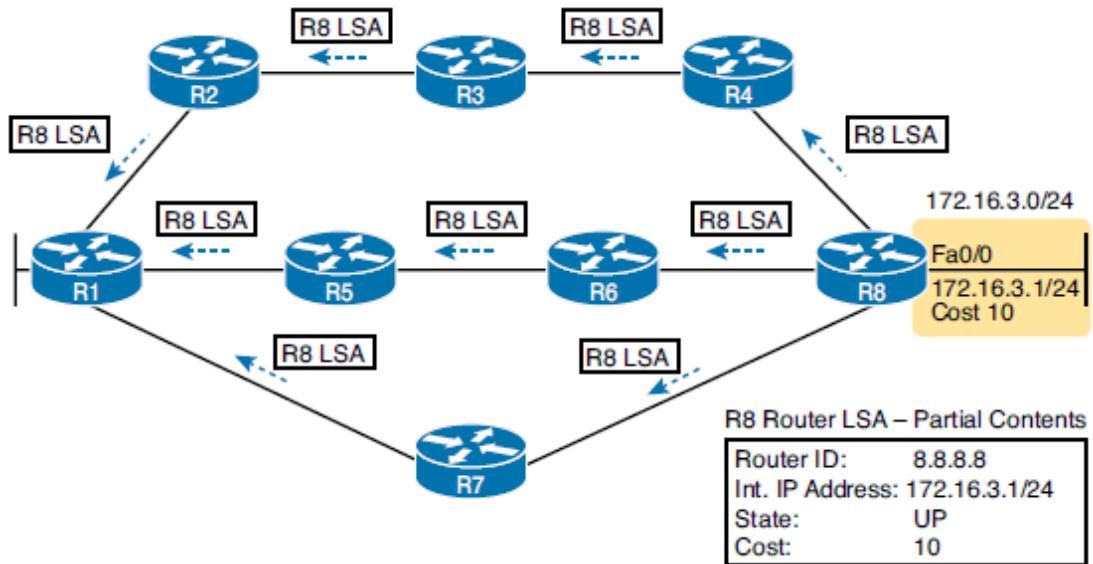
התהליך כולל מספר שלבים:

1. כאשר הנתבים מחליטים להחיליף בינם מידע כדי לסנכרן בינם את בסיס הנתונים שלהם, הם שולחים בנים Database Description (DBD) packet. DBD מאפשר לנtab לדעת האם הבסיס נתונים שלו מסונכרן או שחרר לו מידע.
2. במקרה שחרר לנtab מידע, הוא שולח Link-State Request (LSR) packet. ובתגובה הוא מקבל Link-State Update (LSU) packet.
3. בסוף התהליך הנתבים מסונכרנים בצורה מלאה (Full).



(LSA) Link-State Advertisements

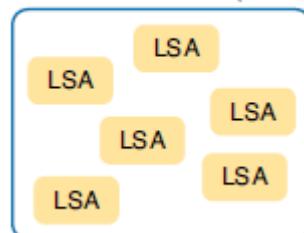
כאשר הטופולוגיה משתנה, הנטב שאצלו השינוי התרחש שולח Link State Packet (LSA) לשכנים שלו כדי לעדכן אותם בשינויו. השכנים שולחים לשכנים שלהם עד שכל הנטבים למדו על העדכון.



(LSDB) Link-State Database

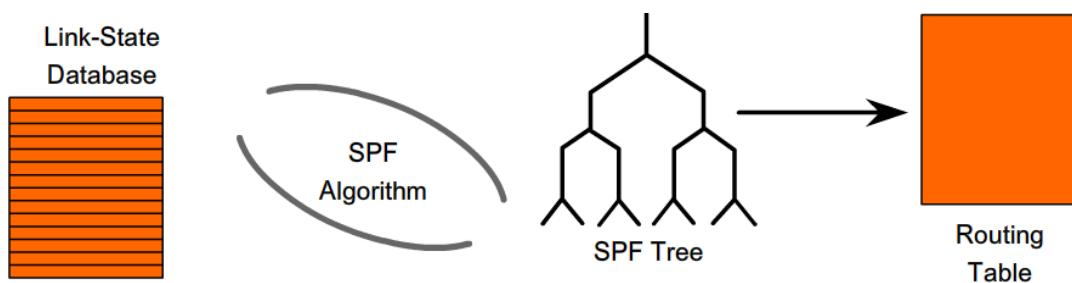
כל ה-LSA packets שנגיעים לנטב יוצרים אוסף של מידע שמתאר את טופולוגיית הרשת ונקרא LSDB. בסוף תהליך העדכון, לכל הנטבים יש את אותו LSDB.

Link State Database (LSDB)



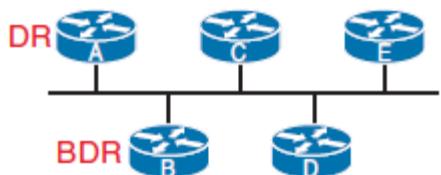
Dijkstra Shortest Path First (SPF) algorithm

SPF זה התהליך שמתבצע על בסיס הנתונים (LSDB) ויוצר את מפת הרשת (SPF tree) מנקודת הראות של הנטב. מפת הרשת משמשת למציאת הנתיב הקצר ביותר ולבנית טבלת הניתוב. בכל פעם שמגיע לנטב Link State packet (LSA), מפת הרשת מחושבת מחדש וטבלת הניתוב מटעדכנת. הנתיבים הטובים ביותר ב-SPF tree, עוברים לטבלת הניתוב.



Designated Router (DR)

בכל Broadcast Domain נבחר נתב אחד שמשמש כ- DR (Designated Router). תפקידו של DR לקלם מהנתבים הודיעות על עדכונים ולפרסם לשאר הנתבים. Backup Designated Router (BDR) מקבל את כל העדכונים וגם DR נופל, BDR מחליף אותו. הנתבים שולחים עדכנים לכתובת 224.0.0.6 (multicast) עליה מאזינים רק DR ו-BDR. DR משדר עדכנים לכתובת 224.0.0.5 (multicast) אליה מאזינים כל הנתבים. הנתב שלממשק שלו מוגדר priority הגבוה ביותר ישמש כ- DR.



אם ה priority בכל הנתבים שווה, אז הנתב עם ה ID Router הגבוה יותר, נבחר להיות DR. הערה: ב- Point to Point Links לא קיימ DR ו-BDR.

ospf priority

בכל Broadcast Domain, הנתב שלממשק שלו מוגדר priority DR. הגובה ביותר ישמש כ- DR והנתב עם ה- priority הבא אחריו יבחר כ-BDR. כבירית מבדיל ה- priority של כל ממשק הוא 1. הערך הגובה ביותר שניתן לתת הוא 255 והוא בעל העדיפות הגבוהה ביותר. הערך 0 קובע שהנתב לא יכול להיות DR או BDR.

```
Router(config-if)#ip ospf priority {number}
```

הגדרת Single Area OSPF

הגדירה הבסיסית כוללת שני שלבים:

- הפעלת פרוטוקול OSPF על הנתב.
- פרסום הרשות שמחוברות לנtab.

הפעלה OSPF

```
router(config)#router ospf {process-id}
```

זה מספר בטוח 1-65,535 שתפקידו לזהות את ה- Process-id OSPF Process השפיעתו מקומית בלבד. ניתן להפעיל כמה OSPF Process באותו נתב וכל אחד מהם פועל בצורה עצמאית ומנהל בסיס נתונים נפרד. כדי לזהות את ה id-Process בצורה קלה, כדי לבחור לכל הנתבים את אותו מספר (נניח 1).

פרסום הרשות

הפקודה מאפשרת לנtab לגלוות דרך איזה מmachים, OSPF ישלח hello packets ואיזה רשות OSPF יפרסם.

```
Router(config-router)#network {network-address} {wildcard-mask} area {area-id}
```

- Network-address - יכול להיות כתובת רשת או כתובת של ממשק בנtab.
0.0.0.0 - אם הכנסנו כתובת של ממשק, ה- Wildcard-mask יהיה 0.0.0.0
- Area-id - במקרה בה יש רק אזור אחד, id יהיה 0.

לדוגמא, כדי לגרום לנtab להפעיל OSPF אוטומטית על כל המmachים:

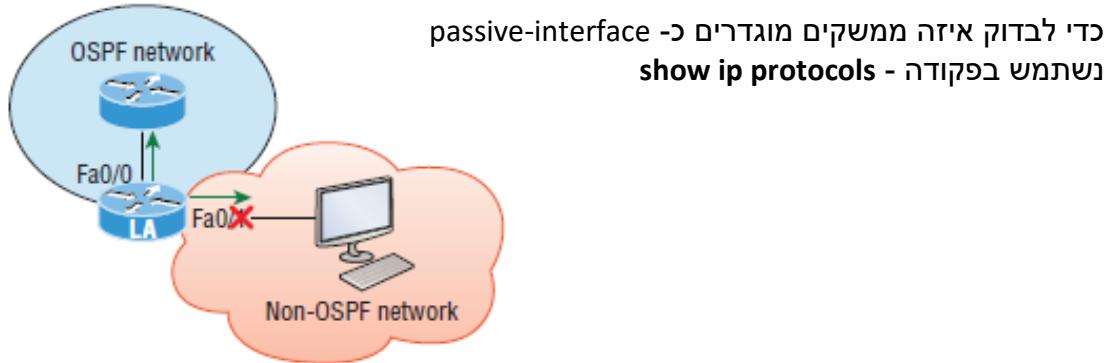
```
Router(config-router)#network 0.0.0.0 255.255.255.255 area {area-id}
```

OSPF Passive Interface

ממשק שמודדר כ- Passive, עושה כך:

- לא שולח או מקבל hello packet דרך הממשק.
- לא מוכן לקייםיחס' שכנות דרך הממשק.

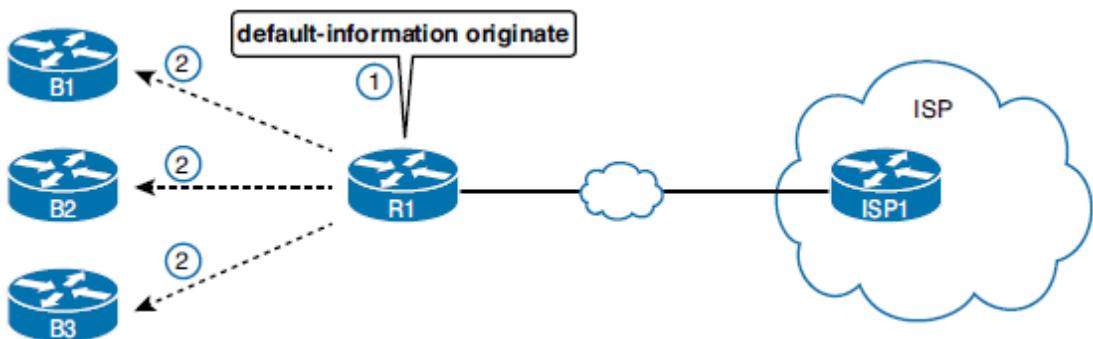
```
router(config)#router ospf {process-id}  
Router(config-router)#passive-interface {fastEthernet 0/1}
```



הגדרת Default Route בסביבת OSPF

הנתב שמודדר בו default route לנット שנמצא מחוץ ל- AS, הוא הנתב שדרכו ניתן לצאת לאינטרנט. נתב זה יכול לפרסם זאת חלק משאר הפרוטוקלים בסביבת OSPF.

```
router(config)#ip route 0.0.0.0 0.0.0.0 {10.0.0.254}  
router(config)#router ospf {process-id}  
Router(config-router)#default-information originate
```



הגדרת (MD5) authentication

כברירת מחדל הפרטוקול OSPF לא משתמש בהצפנה כדי להחליף מידע עם השכנים. זו בעיה אבטחה כי ניתן לצלות לעדכנים ולגלו מידע על הרשת.

.OSPF תומך באימות (authentication) באמצעות Message Digest 5 (MD5).
כך הנתב מוכן לתקשר בצורה מוצפנת רק עם נתבים שמצוירים מולו עם הסיסמה שהוא מכיר (pre-shared password).

:Enable MD5 authentication

```
router(config)#router ospf {process-id}
```

```
router (config-router)#area {area-id} authentication message-digest
```

:Enable OSPF authentication on interface

```
router (config)#interface {interface}
```

```
router (config-if)#ip ospf message-digest-key {key ID} md5 {password}
```

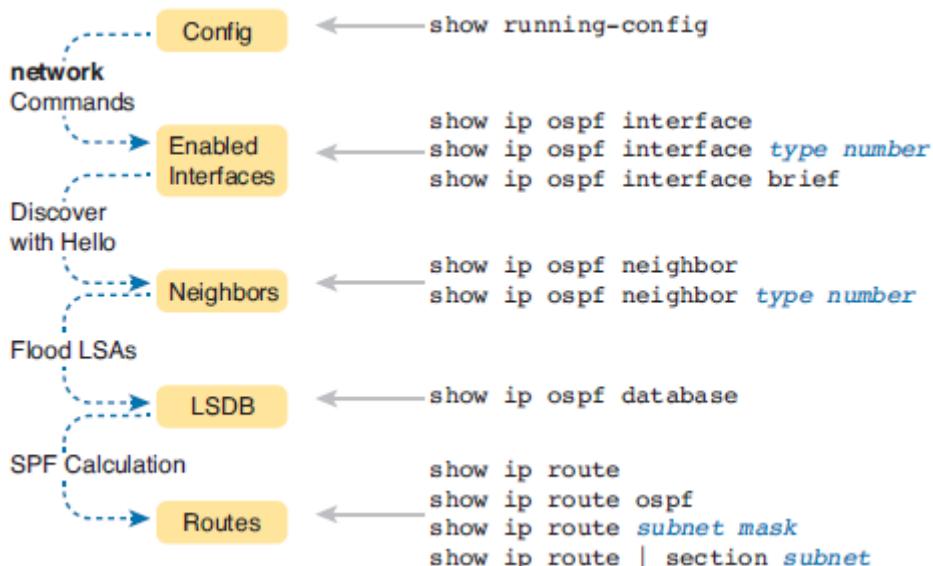
כדי לדעת מה רוחב הפס של ממושך, השתמש בפקודה:

```
Router#show interfaces <fastEthernet 0/0>
```

ניתן להשתמש בפקודה ospf cost כדי לתת עדיפות שונה למושך.

```
Router(config-if)#ip ospf cost <5>
```

בדיקות ההגדרות



show ip protocols

הפקודה מציגה מידע על:

- OSPF Process ID
- OSPF Router ID
- OSFF Area
- מושכים הנ忝ב מפרסם.
- מראה את השכנים.

show ip ospf

הפקודה מציגה מידע על כל ה- OSPF processes שפועלים בנתיב:

- Router ID
- Area information
- SPF statistics

LSA timer information ▪

show ip ospf interface

הפקודה מציגה:

כתובות IP של הממשקים ובאייה Area כל משק.

R1# show ip ospf interface brief								
Interface	PID	Area	IP Address/Mask	Cost	State	Nbrs	F/C	
Se0/0/1	10	0	192.168.10.5/30	15625	P2P	1/1		
Se0/0/0	10	0	172.16.3.1/30	647	P2P	1/1		
Gi0/0	10	0	172.16.1.1/24	1	DR	0/0		

- Process ID ▪
- Router ID ▪
- Cost ▪
- Priority ▪
- DR/BDR ▪

OSPF cost

OSPF משתמש ב-cost במקום ב-metric, כדי לקבוע את הנטייה הטובה ביותר.

ככל שהוא הפס של משק גבוה יותר, כך cost נמוך יותר והנטיב טוב יותר.

כברית מחדל כל מהירות מעל 100Mbps מקבלת ערך של cost (1).

Interface Type	Reference Bandwidth in bps	Default Bandwidth in bps	Cost
10 Gigabit Ethernet 10 Gbps	100,000,000	÷ 10,000,000,000	1
Gigabit Ethernet 1 Gbps	100,000,000	÷ 1,000,000,000	1
Fast Ethernet 100 Mbps	100,000,000	÷ 100,000,000	1
Ethernet 10 Mbps	100,000,000	÷ 10,000,000	10
Serial 1.544 Mbps	100,000,000	÷ 1,544,000	64
Serial 128 kbps	100,000,000	÷ 128,000	781
Serial 64 kbps	100,000,000	÷ 64,000	1562

Same Cost due to reference bandwidth

כדי לסדר זאת יש לשנות את הערך של Reference Bandwidth מ- 100 ל- 10,000. חובה להגדיר זאת בכל הנטים ב-OSPF domain.

R1(config)#router ospf 1

R1(config-router)#auto-cost reference-bandwidth 10000

auto-cost reference-bandwidth 10000

Interface Type	Reference Bandwidth in bps	Default Bandwidth in bps	Cost
10 Gigabit Ethernet 10 Gbps	10,000,000,000	÷ 10,000,000,000	1
Gigabit Ethernet 1 Gbps	10,000,000,000	÷ 1,000,000,000	10
Fast Ethernet 100 Mbps	10,000,000,000	÷ 100,000,000	100
Ethernet 10 Mbps	10,000,000,000	÷ 10,000,000	1000
Serial 1.544 Mbps	10,000,000,000	÷ 1,544,000	6477
Serial 128 kbps	10,000,000,000	÷ 128,000	78125
Serial 64 kbps	10,000,000,000	÷ 64,000	156250

כדי לראות איזה cost מוגדר למעשה, השתמש בפקודה **show ip ospf interface {s0/0/0}**

Enhanced Interior Gateway Routing Protocol (EIGRP)

EIGRP פותח על ידי Cisco והחל משנת 2013 הוא ניתן גם בנתבים של חברות אחרות. הפרוטוקול שיר למשפחת Distance Vector, אך מנצח גם יתרונות של Link State, כמו למשל יישום טופולוגיה הרשת אך הוא שומר נתיבי גיבוי.

EIGRP תכונות

- **Fast Convergence** - התכנסות מהירה, הטופולוגיה מתעדכנת מהר לאחר שינוי וזאת בזכות קיומם של נתיבי גיבוי.
- **Load balancing** - EIGRP מבצע load balance בין נתבים עם cost שווה וזאת כדי למנוע עומס על נתיב מסוים. EIGRP תומך בבחירה load balance בין נתבים עם cost שונה, אך פונקציה זו לא פעילה כברירת מחדל.
- **יעיל** - שולח עדכונים ב- Multicast (224.0.0.10) רק שהטופולוגיה משתנה.
- **קל מאד להגדירה**
- **VLSM** (Variable-Lenth Subnet Masking) - תומך ב- **Classless**

EIGRP טבלאות

מלבד טבלת הניתוב, EIGRP מתחזק שלוש טבלאות:

▪ **טבלת הממשקים**

הטרבלה מכילה את הממשקים בנtab שתומכים ב- EIGRP. כדי לראות את טבלת הממשקים, השתמש בפקודה:

```
Router#show ip eigrp interfaces
```

▪ **טבלת השכנים**

הטרבלה מכילה נתבים שכנים שתומכים ב- EIGRP.

כדי לראות את טבלת השכנים, השתמש בפקודה:

```
Router#show ip eigrp neighbors
```

▪ **טבלת הטופולוגיה**

הטרבלה מכילה את כל הנתיבים שנלמדו מהשכנים.

כדי לראות את טבלת המשקיקים, השתמש בפקודה:

```
Router#show ip eigrp topology
```

טבלת השכנים (neighbors)

שכן זהו נתב שתומך ב- EIGRP וגם מחובר לנtab שלו באותו Broadcast Domain. נתב לומד/מלמד נתבים חדשים רק מנתבים שכנים ולן הקשר עם השכנים מאוד חשוב. כל נתב שולח Hello packet ב- Multicast (224.0.0.10) כל 5 או 60 שניות (בהתאם למירות המשק) וזאת כדי לגלוות שכנים חדשים ולתחזק איתם את הקשר. Hello packet מכיל חומרה זמן שמגדרה כמה זמן הודעה בתוקף (Hold Time). אם עד סוף פרק הזמן (Hold Time) לא נשלח Hello packet חדש, הנתב נמחק מטבלת השכנים.

Bandwidth	Example Link	Hello Interval	Default Hold Time
1.544 Mbps	Multipoint Frame Relay	60 seconds	180 seconds
Greater than 1.544 Mbps	T1, Ethernet	5 seconds	15 seconds

```
Router# show ip eigrp neighbors
```

IP-EIGRP Neighbors for process 77

Address	Interface	Holdtime (secs)	Uptime (h:m:s)	Q Count	Seq Num	SRTT (ms)	RTO (ms)
160.89.81.28	Ethernet1	13	0:00:41	0	11 4	20	
160.89.80.28	Ethernet0	14	0:02:01	0	10 12	24	
160.89.80.31	Ethernet0	12	0:02:02	0	4 5	20	

- ספירה לאחר מכן מחיקת נתב מהטבלה. **UpTime** - כמה זמן השcn בטבלה.
- מספר של packet אחרון שהתקבל. **Seq Num** - Packets - **Q Count**
- הזמן שלקח לחבילה EIGRP להגיע לשcn, ולחזור בחזרה. **SRTT**
(Smooth round-trip time) - הזמן שהנתב יחבר לפני שילוח הودעת Unicast במקורה **Retransmission timeout**) **RTO**
שלא התקבלה תשובה מהשcn.

טבלת הטעופולוגיה

בטבלה מופיעים כל הנתבים שנלמדו מהשכנים.
מטרת טבלת הטופולוגיה, לאחסן נתבי גיבוי (**Feasible Successor**) שימושים כתחליף
במידה ונטי ראי מטבלת הניתוב (**Successor**), לא זמן יותר.
DUAL (Diffusing Update Algorithm) הוא תהליך שמשתמש בטבלת הטופולוגיה כדי לזהות
לכל יעד עד ארבע נתבים ראשיים שונים והם מופיעים גם בטבלת ניתוב.

```
Router# show ip eigrp topology
```

IP-EIGRP Topology Table for process 77

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
r - Reply status

```
P 172.16.90.0 255.255.255.0, 2 successors, FD is 0
    via 172.16.80.28 (46251776/46226176), Ethernet0
    via 172.16.81.28 (46251776/46226176), Ethernet1
    via 172.16.80.31 (46277376/46251776), Serial0
P 172.16.81.0 255.255.255.0, 1 successors, FD is 307200
    via Connected, Ethernet1
    via 172.16.81.28 (307200/281600), Ethernet1
    via 172.16.80.28 (307200/281600), Ethernet0
    via 172.16.80.31 (332800/307200), Serial0
```

P - לא מבוצעים חישובי EIGRP על נתיב זה והוא זמין.
A - מבוצעים חישובי EIGRP על נתיב זה והוא לא זמין.
U (Update) - נשלחה חבילה עדכון לרשות זו.
R (Replay) - נשלחה חבילה Replay לרשות זו.
r (Replay Status) - נשלחה חבילה Query לרשות זו, והנתב מכחילה לתשובה.

מושגים

- נתיב **לא זמין** כי הנטב מחפש נתיב חליפי לנطיב.
- נתיב **זמין**, הנטב לא צריך לבצע חישוב שקשרו לנטיב כי הנטיב יציב.

טבלת הניתוב

בטבלת הניתוב מופיעים רק הנתיבים הטובים ביותר (עד 4 נתיבים לכל יעד).

פינץ EIGRP מהשכזת ה-Metric של נקיין?

EIGRP משתמש בחמש פרמטרים כדי לחשב את ה- Metric של כל נתיב:

- **Bandwidth** (רווח פס) – מהירות השידור דרך הממשק. פרמטר זה נקבע לפי רוחב הפס הנמור ביותר בנתיב. ניתן לשנות ערך זה ידנית לכל ממשק.
- **Reliability** (אמינות) – נקבע לפי אמינות הממשק. פרמטר זה נקבע לפי הממשק עם האמינות הנמוכה ביותר בנתיב. הערך יכול להיות בטוחה 0-255. ערך של 255 מעיד על ממשק תקין ב 100%.
- **Delay** (עיכוב) – נקבע לפי סוג הממשק. פרמטר זה נקבע לפי סכום כל העיקובים בכל הנתיב. ניתן לשנות ערך זה ידנית לכל ממשק.
- **Load** (עומס) – נקבע לפי כמות העומס על הממשק. פרמטר זה נקבע לפי הממשק עם האמינות הנמוכה ביותר בנתיב. הערך יכול להיות בטוחה 0-255 כאשר ערך של 255 מעיד על ממשק עמוס ב 100%.
- **MTU** – הגודל המקסימלי של packet.

כברית מחדל, EIGRP משתמש רק ב- Bandwidth | Delay כדי לחשב את ה Metric של נתיב. הסיבה לכך היא שהעומס והאמינות יכולים להשנות בתדריות גובהה ולכן השינוי של טבלת הניתוב יהיה בתדריות גובהה וזה מוביל לחוסר יציבות. מומלץ לא לשנות את ערכי ברירת המחדל.

Default K Values:

K1 = 1

K2 = 0

K3 = 1

K4 = 0

K5 = 0

כדי לבדוק מה ערכי K, נשתמש בפקודה:

Router#**show ip protocols**

הנוסחה לחישוב ה- Metric :

$$\text{Metric} = [(K1 * \text{Bandwidth} + (K2 * \text{Bandwidth}) / (256 - \text{Load}) + K3 * \text{Delay}) * K5 / (K4 + \text{Reliability})] * 256$$

$$\text{Metric} = [(1 * \text{Bandwidth} + (0 * \text{Bandwidth}) / (256 - \text{Load}) + 1 * \text{Delay}) * 0 / (0 + \text{Reliability})] * 256$$

$$\text{Metric} = (\text{Bandwidth} + \text{Delay}) * 256$$

הנוסחה המוצומצת:

$$\text{Metric} = [(10,000,000 / \text{Min. BW}) + (\text{Sum of Interface Delays} / 10)] * 256$$

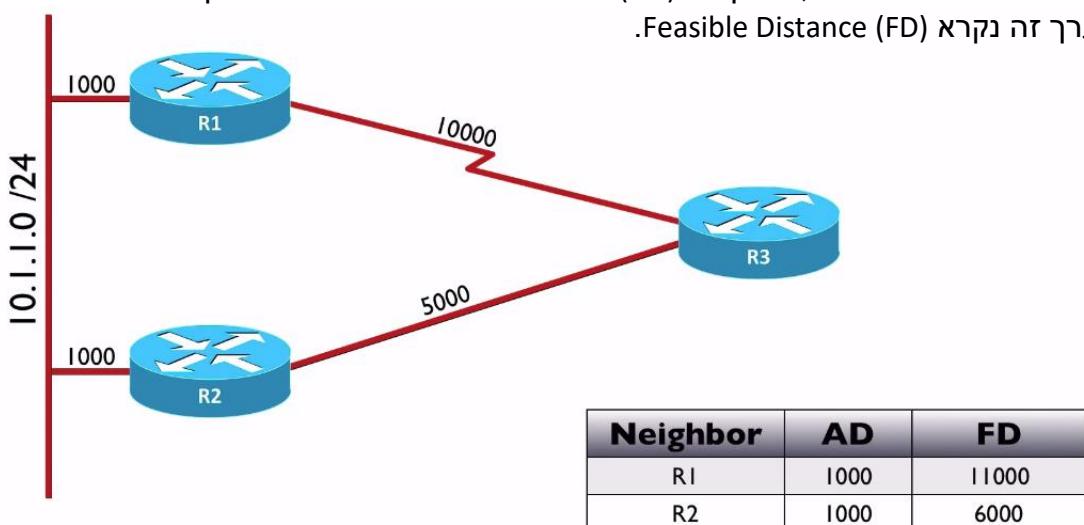
בוחירת הנתיב השטוב בין נתיבים

מה זה (**AD**) ?**Advertised Distance** (AD)

AD זה המרחק (metric value) של הנתיב המפרסם מהיעד המפורסם. כאשר הנתיב מפורסם לשכנים שלו נתיב, הוא מפרסם גם את (AD) של Advertised Distance של הנתיב.

מה זה (**FD**) ?**Feasible Distance** (FD)

FD זה המרחק מהנתיב שלומד את הנתיב ועד ליעד שפורסם בנתיב. הנתיב שלומד את הנתיב, מוסיף לו Advertised Distance את המרחק לננתב המפורסם. ערך זה נקרא (**FD**) Feasible Distance (FD).



יכיז EIGRP יידע מי הוא הנתיב הטוב ביותר ליעד?

בטבלת הטופולוגיה נמצאים כל הנתיבים שהנתיב מכיר. מתוך טבלת הטופולוגיה נבדק מי הוא הנתיב הטוב ביותר לפחות לכל יעד ונתיב זה מועתק לטבלת הנתיב. כדי לחשב מי הנתיב הטוב ביותר, EIGRP משתמש באלגוריתם שנקרא **Diffusing Update Algorithm (DUAL)**.

הנתיב הטוב ביותר הוא הנתיב עם ה- **Feasible Distance (FD)**. לאחר מכן יתבצע נסיעה יתור (Forwarding Successor). לאחר מכן יתבצע נסיעה יתור (Successor). שאר הנתיבים, עם עדיפות נמוכה יותר, נקראים **Feasible Successor** (ירוש). (ירוש אפשרי) והם משמשים כנתיבי גיבוי ונמצאים בטבלת הטופולוגיה.

החוק למניעת לולאות ניתוב:

כדי שנתיב יחשב כנתיב גיבוי, הוא נדרש להיות נמוך יותר מ- FD של Successor (הנתיב הטוב ביותר).

EIGRP Packets בפיפט

- **Hello** - מציאת שכנים ותחזוקת הקשר איתם.
- **Update** - שליחת עדכון, רק במידת הצורך שמתבצע שינוי בטבלת הטופולוגיה. אם מתגלה שכן חדש, נשלחים אליו update packets שמעבירים לו את כל טבלת הטופולוגיה. בכל מקרה, נתיב מפורסם רק את הנתיב הטוב ביותר ליעד.

- **Query** (בקשה) – אם נתיב הופך ללא זמין, הנטב שולח **query packet** במטרה לחפש נתיב חליפי.
- **Reply** (תגובה) - נשלח כתגובה ל- **Query**.
- **Ack** (אישור) - נשלח אישור על קבלת **Query**, **Update** או **Reply**.

EIGRP

הגדירה כוללת שני שלבים:

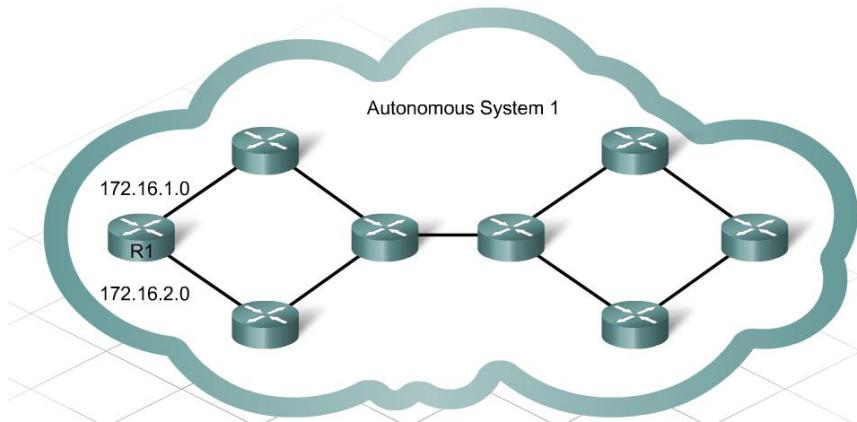
הפעלת הпрוטוקול תוך קביעת system autonomous
 autonomous system זהו מספר שמצוה את כל הנטבים ששיכים אותה קבוצה.
 המספר יכול להיות בטוחה 1-65535 וא| ריך להיות זהה בכל הנטבים באותה קבוצה. |

```
R1 (config) #router eigrp {1}
```

פריטם הרשות שמחוברות פיסית לנטב
 עליו להחליט על איזה מושגים הprotocole EIGRP יהיה פעיל ואיזה רשותה הנטב יפרעם.
 כאשר אנו משתמשים בפקודה **network**, הנטב בודק האם קיימים בנטב מושגים שתואימים
 לכתובת שהוכנסה. אם נמצאו מושגים תואמים אז הנטב יפרעם רק אותם.

לדוגמא:
 הגדרה זו בנטב R1, תפרעם שני כתובות רשת (172.16.2.0 | 172.16.1.0).

```
R1 (config-router) #network {172.16.0.0} {0.0.255.255}
```



הערה: בפריטם רשת, אין חובה להכניס Wildcard masks. אם מוכנו Class-full Wildcard masks מסווג Class-full.

Auto Summarization

כברירת מחדל, EIGRP מבצע auto summarization, כלומר כאשר הנטב מפרסם רשותה, הוא מפרסם אותן כ-classful.

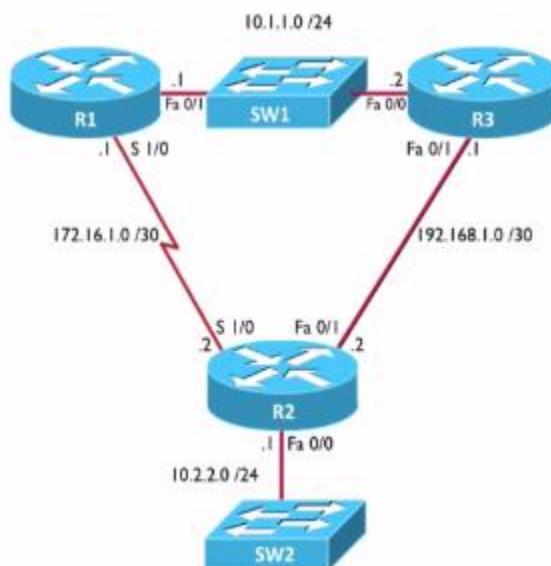
בדרכ כל זה דבר טוב כי זה מצמצם את טבלת הניתוב אבל אם הרשות בינויו לצורך לא אחידה זה יכול לגרום לביעות ניתוב.

כדי לבטל את תcona זו:

```
R1 (config-router) # no auto-summary
```

כדי לבדוק האם auto summarization פועל, השתמש בפקודה:

```
Router#show ip protocols
```



Passive Interface

הגדרה זו מונעת ממשק לשדר ולקבל עדכנים. מטעמי אבטחת מידע, עדיף למנוע ממשק לשדר ולקבל עדכנים, אם דרך ממשק זה אין לשכנים.

לדוגמא: בתמונה לעיל, אין צורך לשלוח עדכנים דרך דרך ממשק 0/0/0.

```
Router (config-router) #passive-interface { fastEthernet } { 0/0 }
```

כדי לבדוק האם קיימים ממשקים במצב Passive, השתמש בפקודה:

```
Router#show ip protocols
```

האגרת **authentication** (האגרת רשת האבטחה)

כדי להגבר את רמת האבטחה, רצוי לזרות את השכן שלנו כתוב מורה. לשם כך אנחנו משתמשים בתהילך הזדהות בין הנתבים.

טהילך הגדרת **authentication** כולל שני שלבים:

1. יצירת המפתח: (הסימא צריכה להיות זהה בכל הנתבים)

הסימא נחתמת דיגיטלית באמצעות פרוטוקול MD5 והחתימה הדיגיטלית נשלחת בין הנתבים.

```
R1(config) #key chain {name-of-chain}  
R1(config-keychain)#key {key-id}  
R1(config-keychain-key)#key-string {password}
```

2. הפעלת ההזדהות:

```
R1(config-if)#ip authentication key-chain eigrp  
{autonomous system parameter} {name-of-chain}  
  
R1(config-if)#ip authentication mode eigrp  
{autonomous system parameter} md5
```

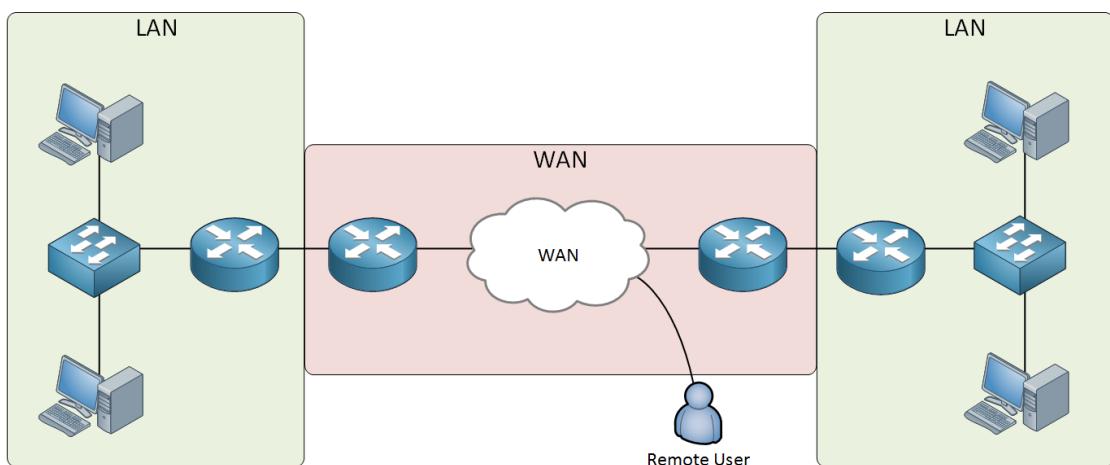
כדי לראות את הסימא, השתמש בפקודה: (הפקודה לא נתמכת ב-
(Packet Tracer) R3#show key chain

Wide Area Networks (WAN)

הרשתות שלמו נקראות LANs (רשת תקשורת מקומית). רשתות אלו בבעלותנו ומתחזקים אוטם. כל התקנים שמרכיבים את הרשת LAN קרובים זה לזה, בבניין אחד או כמה בניינים קרובים זה לזה.

כאשר אנו צריכים לגשת לרשתות מרוחקות, לחבר שתי רשתות LAN או לנתן לאחרים גישה לרשת המקומית שלנו, אנו זקוקים לרשת WAN. רשת WAN מכסה אזורי גיאוגרפיים גדולים כמו רשת בין ערים או כמו האינטרנט.

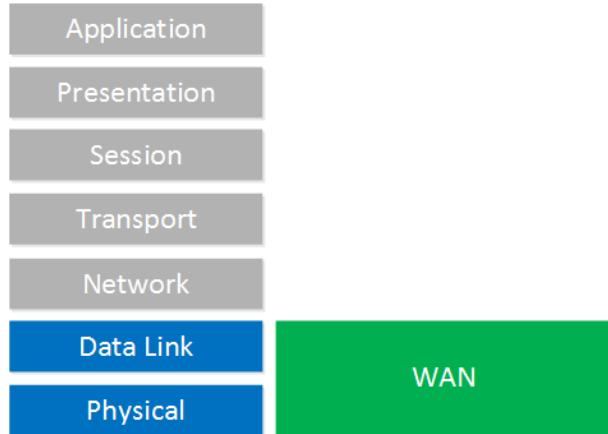
רשתות WAN מופעלות על ידי חברות כמו חברות טלפון/כבלים, ספקי שירות או חברות היי-אן. הם בונים רשתות גדולות שמתפרשות על ערים או אזוריים שלמים ומחקרים את הזכות להשתמש ברשתות שליהם לקוחות שלהם.



ב- LAN, הפרוטוקול הדומיננטי שבו אנו משתמשים הוא Ethernet. בעבר WAN, יש שירותים טכנולוגיים ופרוטוקולים שאנו יכולים לבחור.

כל הצד תקשורת והפרוטוקולים שקשרים ל-WAN, שייכים לשכבה הפיסית ולשכבה קישור הנתונים במודל OSI. בשכבה הפיזית, אנו משתמשים בחומרה, כבלים שונים, מחברים וממשקים. בשכבה קישור הנתונים, ישנו מספר פרוטוקולי WAN שונים בהם אנו יכולים להשתמש.

OSI Model



WAN Topology

טופולוגיה פיזית מתארת את הפריסה הפיזית של הרשת, בנווגוד לטופולוגיה לוגית שמתארת את הנטיב שאות עובר דרך הטופולוגיה הפיזית.

ישנם שלושה טופולוגיות בסיסיות לתכנון WAN:

- hub-and-spoke Star
- Fully meshed
- Partially meshed

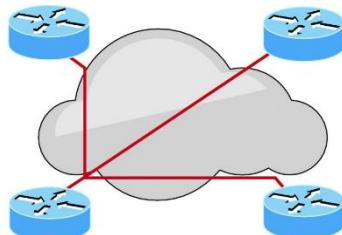
(hub-and-spoke) Star

טופולוגיה זו מספק גישה מרשתות מרוחקות לנット ליבה (ネット 中心) כל התקשרות בין הרשותות עוברת דרך נット הליבה.

ה יתרונות של טופולוגיה כוכב פיזית הם פחות עלות ניהול כל יותר, אבל החסרונות יכולים להיות משמעותיים:

- הנット המרכז מייצג נקודת כשל מרכזית.
- הנット המרכזי מגביל את הביצועים של גישה למשאים מרכזיים.

זה צינור יחיד שמנהל את כל התנועה המיועדת למשאים המרכזיים או לנטים אחרים.

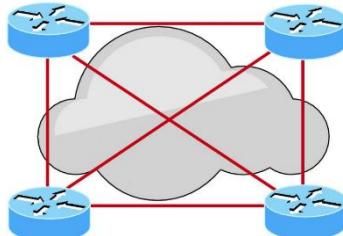


Fully meshed

טופולוגיה זו, לכל נット יש נתיב ישיר לכל נット אחר בענן. טופולוגיה זו מספקת בבירור רמה גבוהה של יתירות, אך העליות הן הגבהות ביותר ולקן טופולוגיה Fully meshed לא קיימת ברשת גדולה. הנה כמה בעיות בטופולוגיה זו:

- עלות גבוהה.

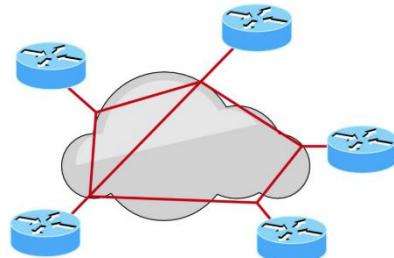
- הקונFIGורציה מורכבת יותר.



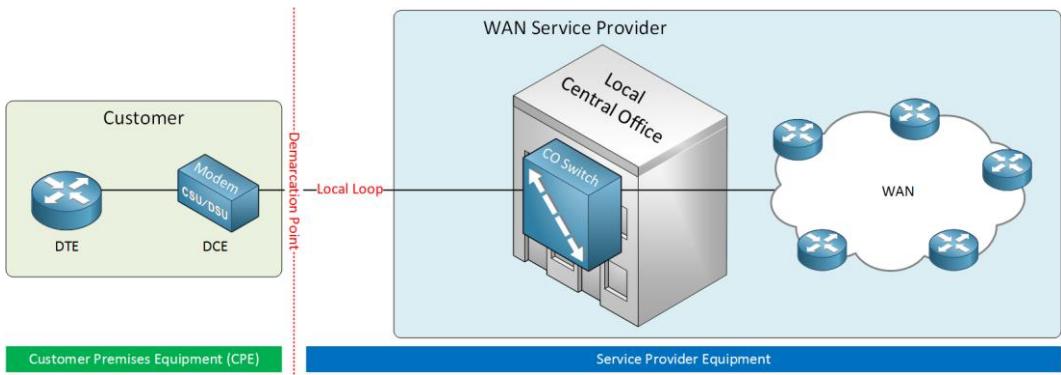
Partially meshed

סוג זה של טופולוגיה מקטין את מספר הנטים בתוך הרשת שיש להם קשרים ישירם לכל נתבים אחרים בטופולוגיה.

שלא כמו בטופולוגיה Fully meshed, כל הנטים אינם מחוברים לכל נתבים אחרים, אבל זה עדין מספק יתרונות יותר מאשר טופולוגיה Star. למעשה זו נחשבת לטופולוגיה המאוזנת ביותר.



Physical Layer Terminology



CPE (Customer Premises Equipment) - זה הציון שנדרש כדי לחבר את הלוקו לספק שירותי WAN. בדרך כלל הציון כולל: נתב בעלות הלוקו ומודם שמושכר מספק השירות.

DTE (Data Terminal Equipment) - זה התקן הלוקו שמעביר נתונים מרשת הלוקו לרשת WAN. זה יכול להיות נתב, מחשב או מtag.

DCE (Data Circuit-terminating Equipment) - זה מכשיר שמקבל נתונים מי WAN בצורה דיגיטלית ומתרגם אותם לצורה אנלוגית (מודם) או דיגיטלית מותאמת לו. לעתים DCE משולב בנATAB (CSU/DSU).

CSU/DSU (Channel Service Unit/Data Service Unit) - זה התקן שיושב בין הנתב לבין WAN. הוא ממיר אותות דיגיטליים מה-WAN אל אות דיגיטלי שהנתב מבין ולהיפך. לדוגמה, CSU/DSU יכול להיות מחובר לנatab באמצעות כבל טורי (DTE), لكن עליו לדבר בשפה שנATAB מבין. הצד השני, ה-CSU/DSU מחובר לשופק שירותי WAN שմדבר בשפה אחרת.

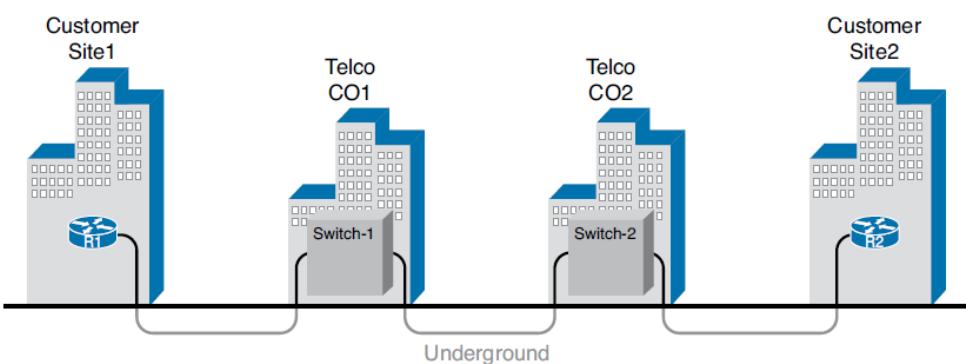
בעבר, CSU/DSU היה מכשיר נפרד אך כיום הוא משולב במכשיר הנתב.

Demarcation Point (نקודת התיאום) - זה המקום שבו חיווט ספק השירות מסתיים ווחיווט של הלוקו מתחיל. זה יכול להיות ארון בתוך או מחוץ לבית שלך או הבניין.

Local Loop (Demarcation Point) - היא הקישור הפיזי שמתחבר מנקודת התיאום (Demarcation Point) של הלוקו לקצה רשת ספק השירות.

CO (Central Office) - זה הבניין אליו מתחברים כל ה-Local Loops מהлокאות.

CO זהה המקום אליו מחובר הלוקו וממנו ניתן לחבר את הלוקו לשאר היעדים. בדרך כלל המרחק של הלוקו מי- CO מושפע על המהירות מקסימלית שהוא יכול לקבל מהספק שירות. בבנין זה, אנו מוצאים CO switches שככל הלוקו שמשתמש מתחברים אליו. הסוג של המתגים תלוי בטכנולוגיה שהספק שירות משתמש (טלפוניה, DSL, כבל...).



WAN Technologies

ישנן מספר שיטות שונות כיצד ניתן להעביר נתונים ברשת:

- Circuit switching
- Packet switching
- Cell switching

Circuit Switching (MITOG מעגליים)

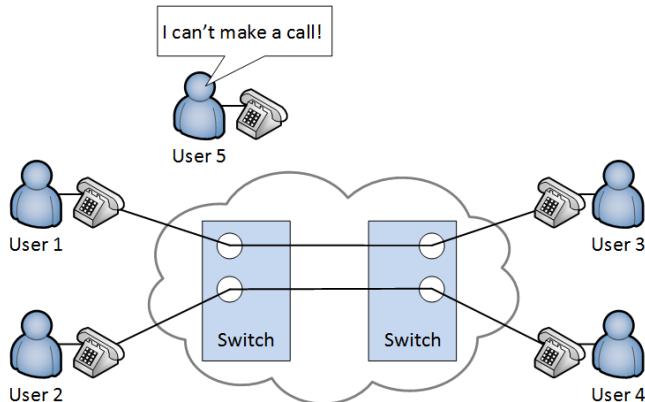
זה פתרון זול ווישן מאד שטבoso על תשתית הטלפוניה (המודם של פעם).
PSTN (Public Switched Telephone Network)

בטכנולוגיה זו, נוצר חיבור ישיר רק בזמן שידור נתונים. למרות שיש דרכים רבות להגעה
מןקוודה לנקוודה, נבחרת דרך אחת וכל התקשרות עוברת בדרך זו.

כיוון טכנולוגיה זו משמשת בעיקר גיבוי.

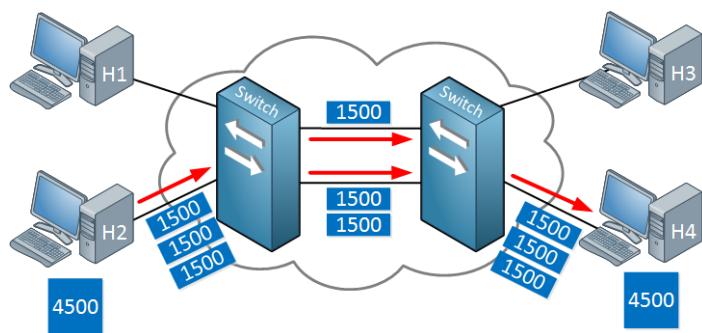
טכנולוגיות שימושísticas בשיטה זו:

- Dial-up Modems
- .kbps 64 – ISDN (Integrated Services Digital Network)



Packet Switching (MITOG מנות)

הרעילון מאחורי Packet Switching זה שאנו שוברים את הנתונים שלנו לחטיכות.
כל חטיכה היאmana שנשלחת ברשת. זו השיטה העיקרית בימינו לשילוח מדע ברשת.

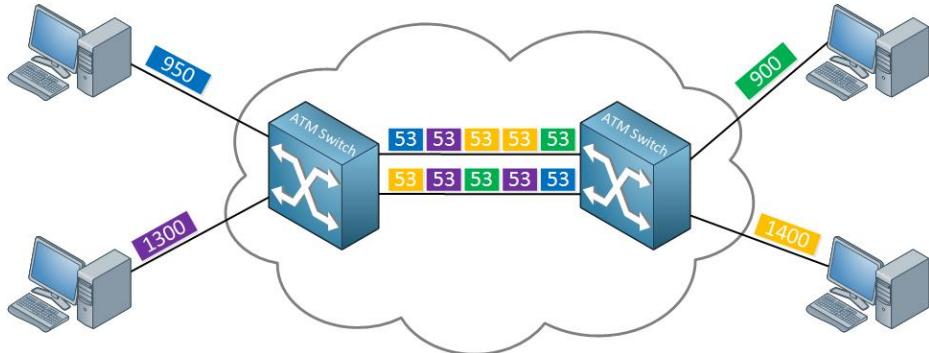


בדוגמא, מחשב 2 רוצה לשלוח 1500 Bytes של נתונים. הוא שובר אותם לשלווש חטיכות של 1500 Bytes. מנה אחת נשלחת על הקישור העליון, שני האחרים על הקישור התחתיון. עם MITOG מנות, אין נתיב קבוע ברשת. לאחר שמחשב 4 מקבל את כל מנות, הוא מחלץ מחדש את הנתוניים. גודל המנה משתנה, אין גודל קבוע.

MITOG מנות החליף בעיקר את MITOG מעגליים. אחד ה יתרונות הוא שnitן לנצל בצורה יעילה יותר את רוחב הפס שיש לרשות להציג וכך אנחנו לא מبذבזים משאביים שאין בהם שימוש.

Cell Switching (מייתוג תאים)

מייתוג תאים דומה מאוד למייתוג מנות למעט שהוא משתמש בגודל קבוע עבור הגודל של המנה.



בדוגמה, ניתן לראות שככל מחשב שולח כמות אחרת של נתונים.

הميدע שהם שלוחים מחולק לתאים בגודל קבוע, 53 bytes בדוגמה זו.

זה פרוטוקול ATM (Asynchronous Transfer Mode) שהשתמש במייתוג תאים.

WAN Technology

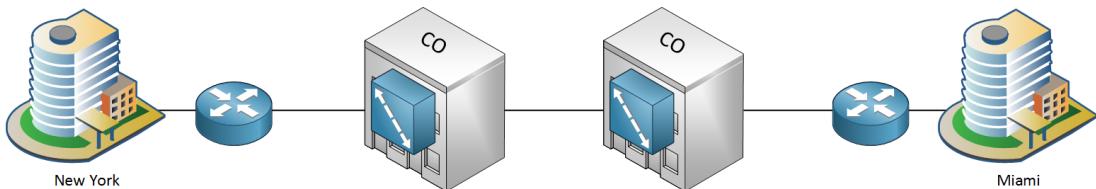
Leased Lines

קוויים מושכרים הם אחת מהאפשרויות ה-WAN הישנות. תארו לעצמכם שיש לנו LAN בניו יורק ועוד LAN במיאמי. אנחנו צריכים לחבר את שתי הרשתות.

קו מושכר הוא חיבור מסווג point-to-point שמצוע לעתים קרובות על ידי חברת טלפונים. הקוו פרטני ו乐观 אמיןותו גבוהה מאוד ומהירות החיבור מובטחת. המחיר נקבע לפי המרחק ומהירות החיבור בין שני האתרים.

- Leased Lines קיימים מספר שמורות:

- T1 / T3 / E1 / E3
- Point-to-point link
- Serial link
- Leased circuit



בשכבה קישור הנטונים, ישנו שני פרוטוקולים שאנו יכולים להשתמש על

- HDLC (High-Level Data-Link Control)
- PPP (Point-to-Point Protocol)

Frame Relay

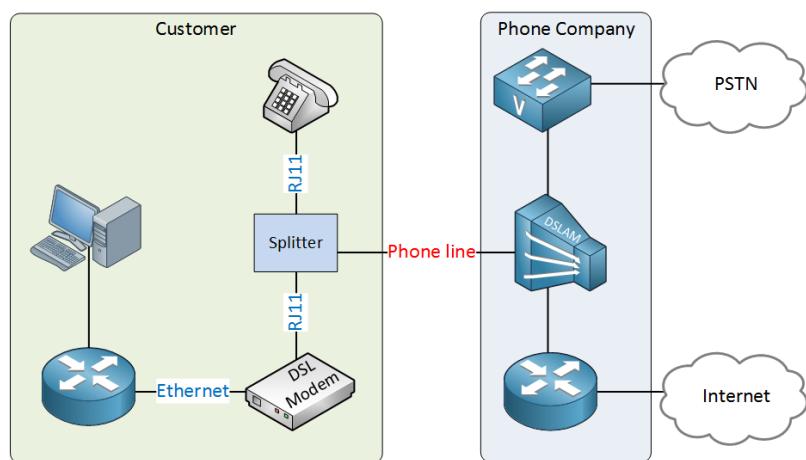
זו טכנולוגיית WAN ישנה שמשמשת כחלופה ל-leased lines.

הטכנולוגיה פועלת באמצעות מייתוג מנות, לחיבור התקנים בראשת (WAN).

ב-leased lines אתה היחיד שמשתמש בקישור ולכן זו אופציה יקרה. Frame Relay מציע leased lines עם רשות משותפת וכן זו אפשרות זיהלה יותר מאשר leased lines.

(Digital subscriber line) **DSL**

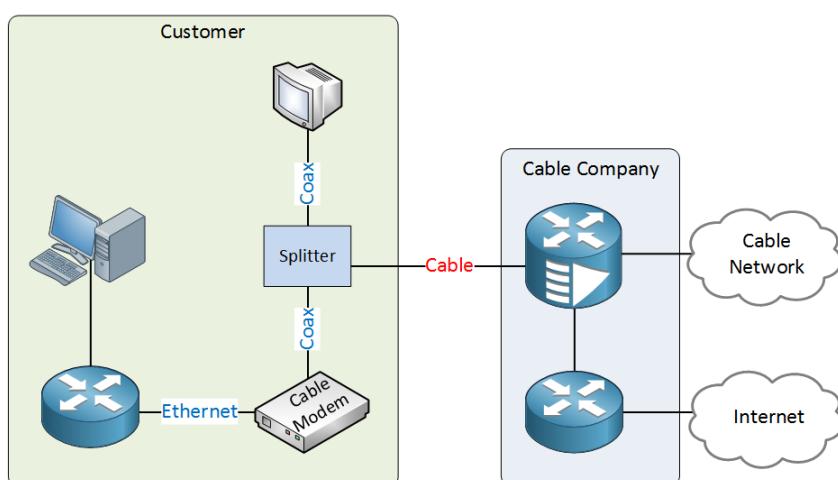
DSL הפרק להיות אופציית פופולרית מאוד עבור גישה לאינטרנט גבוהה שכן הוא משתמש בכבלים טלפון אנלוגי שיש בכל בית או בניין. מהירות שאתה מקבל תלוי מרחק בין הבית שלך לבין חברת הטלפון.



חברת הטלפונים משתמשת במכשור הנקרא DSL (מרובב גישה DSL) אשר מפצל את תנועת הנתונים ואת התנועה הקולית אחד מהשני. תנועת נתונים מועברת לנットב, תנועה קולית למתקן קול. רוב ספקי DSL מציעים מהירות אסימטריות (ADSL), רוחב הפס במורוד הזרם גבוה יותר מאשר רוחב הפס במעלה הזרם.

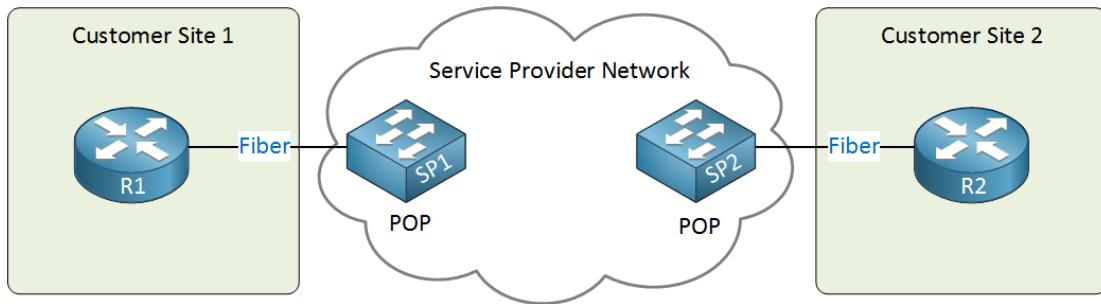
Cable

אינטרנט בכבלים דומה ל-DSL, וגם הפרק פופולרי מאז שברוב הבתים והمبנים יש חיבור כבלים. כבל האינטרנט משתמש בטכנולוגיה שמאפשרת לו להעביר נתונים על גבי כבל קוואקסיאלי. בסביבה מודרנית, הcabלים עובדים בסביבה HFC (hybrid fiber-coaxial), קלומר סיבים אופטיים בשילוב עם כבלים אינטראנט בכבלים לעיתים קרובות מציעה רוחב פס גבוה יותר אבל זה תלוי במספר המנוויים ברשת.



Ethernet

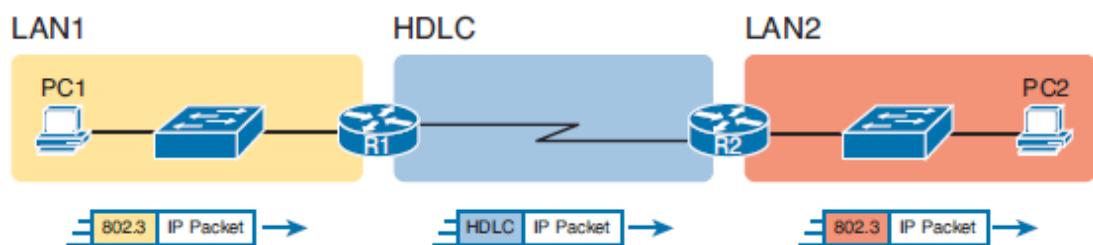
Ethernet גם עשה את דרכו אל WAN. למשל, תקן X-1000BASE-ZX תומך למרחוק של בערך 40 ק"מ על גבי סיב אופטי. מנוקודת המבט של הלקוח, זה דומה לו הוכח.



באתר הלקוח, קיימן נתב עם חיבור סיב אופטי לספק שירות. החיבור בצד ספק השירות נקרא POP (נקודות נוכחות). ספקים רבים קוראים לזה קוו Ethernet פרטי. כמו כן, ניתן לקבל יותר משני אתרים, יצירת רשת מרובת גישה.

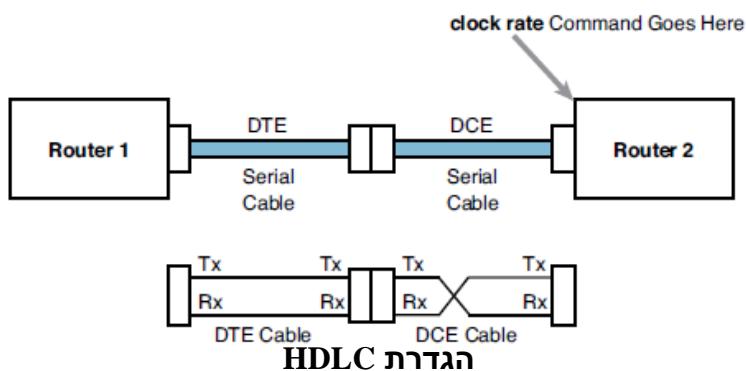
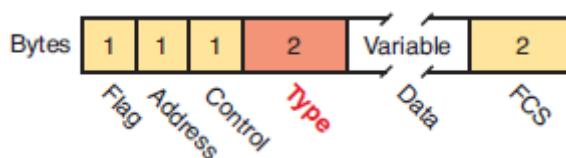
(High-Level Data-Link Control) HDLC

זה point-to-point פרוטוקול שמשמש כפרוטוקול ברירת המחדל ב망ים סריאליים בין נטבבים של סיסקו. יכול יצור תקן HDLC ייחודי ולכך משני צדדי הצלב צריכים להיות נטבבים של אותו יציר. פרוטוקול לא תומך ב-authentication. HDLC מארגן את הנתונים לתוך מסגרות לפני השידור.



מבנה HDLC Frame

Cisco הוספה את השדה Type שמאפשר לשדר סוגים HDLC שונים (לדוגמא: IPv4 או IPv6). השדה Frame Check Sequence (FCS) משמש לבדיקת תקינות ה-frame. כיום השדות Control ו-Address לא בשימוש.



שלב ראשון - הגדרת כתובת לממשק

```

R1(config)#interface serial {0/0/0}
R1(config-if)#ip address {ip} {subnet mask}
R1(config-if)#no shutdown
R1(config-if)#clock rate {2000000}
R1(config-if)#encapsulation {hdlc}

```

שלב שני - הגדרת Clock Rate בэмישק

שלב שלישי - הגדרת Encapsulation

בדיקות ההגדרות

```

R1# show controllers serial 0/0/0
Interface Serial0/0/0
Hardware is SCC
DCE V.35, clock rate 2000000

```

בדיקה האם הэмישק הוא DTE או DCE.
בדיקה של Clock rate.

```

R1# show interfaces s0/0/0
Serial0/0/0 is up, line protocol is up
Hardware is WIC MBRD Serial
Description: link to R2
Internet address is 192.168.2.1/24
MTU 1500 bytes, BW 1544 Kbit/sec, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation HDLC, loopback not set

```

האם הэмישק משתמש ב-HDLC Encapsulation:

(Point-to-Point Protocol) PPP

זהו פרוטוקול WAN מאוד נפוץ ונחשכ כסטנדרט בתעשייתו. עובד בשכבה שנייה של מודל OSI.

תכונות הפרוטוקול:

- בNetworking ל-PPP יכול ליצור חיבורויות בין נתבים של יצרנים שונים.
- תומך בתקשורתynchronous ו-asynchronous.
- תומך בהזדהות מסוג CHAP ו-PAP.
- תומך בדוחינת נתוניים.
- תומך ב- Multi links (איחוד כמה חיבורים לחיבור לוגי אחד).

הגדרת חיבור מסוג PPP

```
R1(config-if)#encapsulation {ppp}
```

הגדרת אימות והצפנה מסוג CHAP:
עלינו ליצור בנתב חשבון משתמש:

```
R1(config)#username {R1} password {123}
```

עלינו להפעיל את האימות על הэмישק:

```
R1(config-if)#ppp authentication chap
```

Layer 3 Redundancy Protocol - HSRP

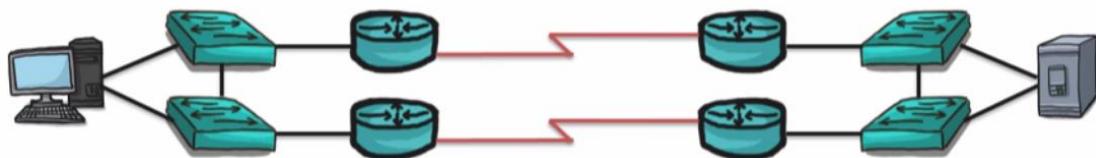
הבעיה

ל-Client Default Gateway אחד ולכון אם יש בעיה בנתב או בקשרו של הנתב לרשתות אחרות, אז Client ברשות לא יכולם לתקשר עם רשתות אחרות.



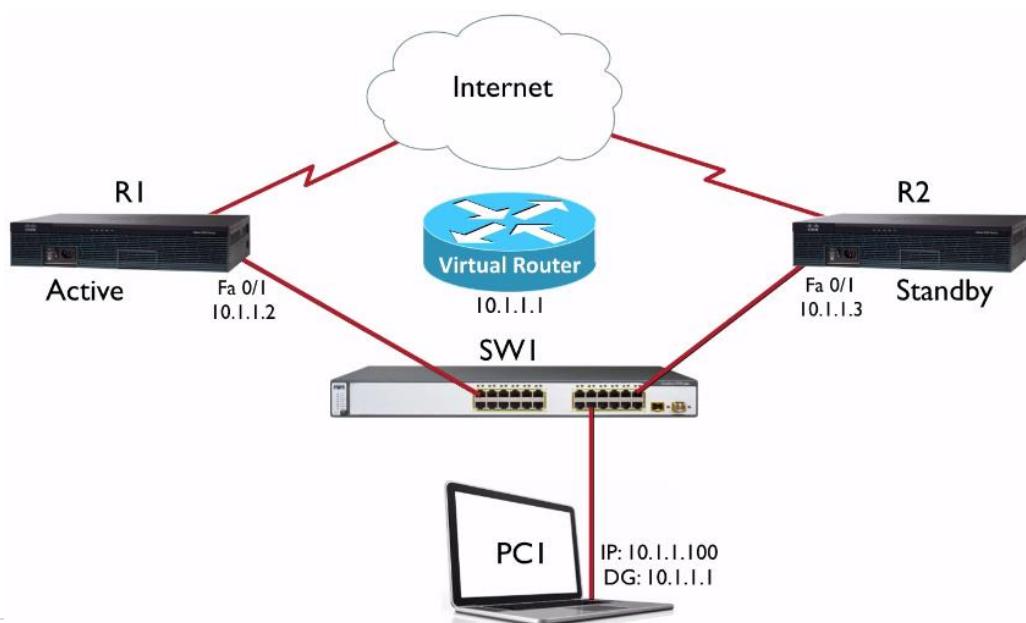
הפתרון

FHRP (First Hop Redundancy Protocols) מאפשר להתקין ברשת מספר נתבים שפועלים כנתב אחד. התקן ברשות לא מודע למורכבות זאת וראה את הנתבים כנתב אחד.



בפועל, FHRP מתאר את שלושת ה프וטוקולים הבאים:

- **HSRP (Hot Standby Router Protocol)**
הפרוטוקול נוצר על ידי cisco בשנת 1994 ומשתמשים בו גם כיום.
- **Load Balancing**
הפרוטוקול לא תומך ב-.
- **VRRP (Virtual Router Redundancy Protocol)**
זהו פרוטוקול שנוצר בשנת 1999 בסטנדרט של IEEE. הפרוטוקול עובד בצורה מואוד דומה ל- HSRP אך יכול לפעול בננתבים של חברות שונות.
- **Load Balancing**
הפרוטוקול לא תומך ב-.
- **GLBP (Gateway Load Balancing Protocol)**
נוצר על ידי cisco בשנת 2005. הפרוטוקול תומך ב- Load Balancing (חלוקת עומסים). כלומר במקום שתעבורת הרשת תעבור רק דרך הנתב הפעיל, תעבורת הרשת עוברת דרך כל הנתבים שבקבוצה.



HSRP (Hot Standby Router Protocol)

בשיטה זו, קבוצת נתבים (Standby Group) מיווצרת לפני הרשת כנתב יրטואלי אחד באמצעות כתובת וירטואלית אחת. הכתובת הוירטואלית משמשת כ- Default Gateway לכל המחשבים ברשת.

נתב אחד מתוקן הקבוצה יתפקיד כנתב פעיל (Active) ושאר הנתבים יהיו בהמתנה (standby). אם הפעיל קורס, אחד מהנתבים יחליף אותו ויעבור מ מצב המתנה להיות הנתב הפעיל וכך ה- Default Gateway יישיר להיות זמין למחשבים ברשת.

מוגדר כברירת מחדל של 3 שניות נשלוח Hello Packet בין הנתבים במטרת לבחור Active Router.

Hello Packet - מוגדר כברירת מחדל ל- 10 שניות. ככלומר, אם לא נשלח Hello Packet במהלך 10 שניות אז הנתב שמתפקידו Standby, הופך להיות Active.

v2 HSRP מאפשר לשנות את ה- Timers

v1 HSRP v1 שולח הודעות ב- udp port 1985 224.0.0.2 multicast לכתובת IP

v2 HSRP v2 שולח הודעות ב- udp port 1985 224.0.0.102 multicast לכתובת IP

אין תאימות בין v1 v2 HSRP ל-

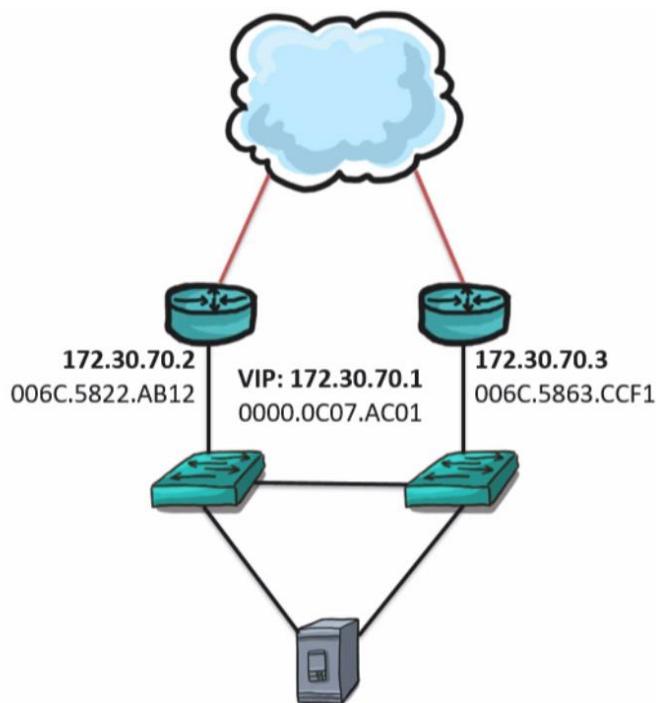
כתבת פיסית וירטואלית

כasher לקויה פונה ל- Default Gateway הוירטואלי, הוא משתמש ב프וטוקול ARP כדי לתרגם את הכתובת IP הוירטואלית לכתובת פיסית.

הנתב שמתפקידו Active ישלח ב- Broadcast הודעת (GARP) Gratuitous ARP (GARP) מודיע על רשותה את הכתובת הפיסית של הרכטים וכן כל המתגים והתקנים יודעים על מיקומו.

הכתובת הפיסית ב- v1 HSRP היא: **0000.0C07.ACxx** (xx זה מספר הקבוצה).

הכתובת הפיסית ב- v2 HSRP היא: **0000.0C9F.Fxxx** (xxx זה מספר הקבוצה).



הגדרת HSRP

הפעלה HSRP

שיות משק ל- 1-255 Standby Group (1-255) ולכטוב וירטואלית:

R1(config-if)#standby {1} ip {192.168.1.1}

הגדרת Active Router

.Active Router עם ה- **הגובה** ביותר נבחר להיות הנטב

:**Priority** Default Priority (100) priority של המשק (100)

R1(config-if)#standby {1} priority {105}

כדי לראות פרטים:

R1#show standby brief

P indicates configured to preempt.							
Interface	Grp	Pri	P	State	Active	Standby	Virtual IP
Gig0/0	1	110	A	Active	local	192.168.1.3	192.168.1.1

P indicates configured to preempt.							
Interface	Grp	Pri	P	State	Active	Standby	Virtual IP
Gig0/0	1	100	S	Standby	192.168.1.2	local	192.168.1.1

Interface Tracking

הfonקציה **Tracking** מאפשרת לעקב אחר מצב של משק מסוים וכך לאתר נתב שמשמש כ- Active אר המשק החיצוני שלו לא זמין. אם המשק לא מתפרק אז מופחת ערך מסוים מי ה- Priority של הנטב וכך נתב אחר מתפרק כ- Active Router

הגדרת Tracking על משק (כברירת מחדל מוריד 10):

R1(config-if)# standby {1} track {gigabitEthernet 0/1} {decrement-value}

Packet Tracer לא נתמך ב- decrement-value

Preempt Option

כברירת מחדל, נתב שהוא Active Router והפוך ל- Standby Router, לא יחוור אוטומטית להיות .Active Router Preempt Option מאפשרת לנטב לחזור להיות活性 Router

הפעלה Preempt Option על משק:

R1(config-if)#standby {1} preempt

שיטתי טים

כדי לשנות את ה- Timers

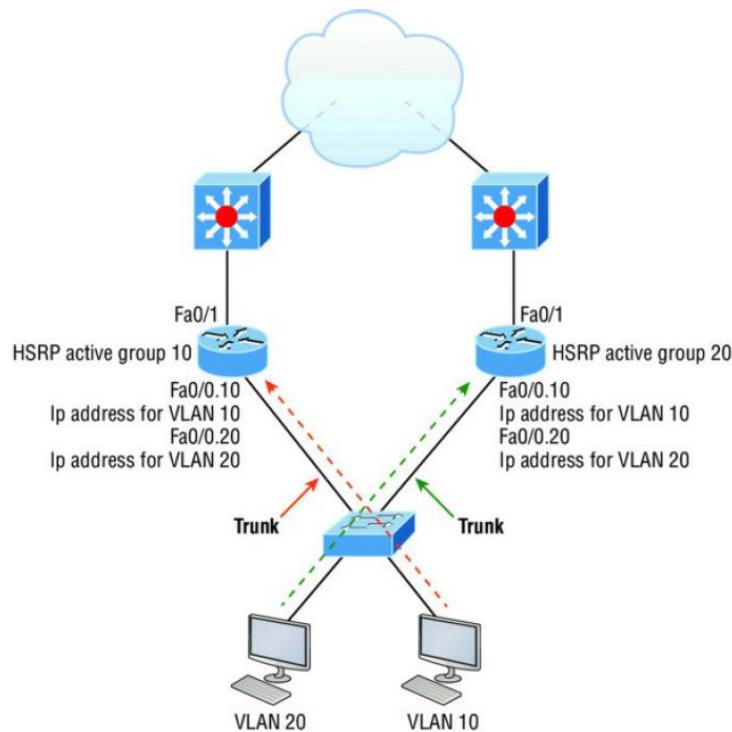
R1(config-if)#standby {1} timers msec {200} mse {700}

Debug (לא לשכוח לכבות בסוף השימוש):

R1#debug standby

HSRP Load Balancing

בדרכם כלל, בהגדירה מתקדמת, לא נשימוש ב- HSRP לאיזון עומסים אלה נשימוש ב- GLBP. HSRP לא באמת ידוע לבצע Load Balancing אבל הוא יכול להיות מוגדר לשימוש ביותר מנתב אחד בכל פעם בשימוש עם VLAN שונים.



HSRP Troubleshooting

רוב הבעיות עם HSRP ניתנות לגילוי עם הפקודה `show standby`. בפלט הפקודה ניתן לראות את:

- Active IP
- MAC address
- Timers
- Active Router
- Standby Router

בדרכם כלל תקלות מגיונות בגדרה לא נוכנה של הנושאים הבאים:

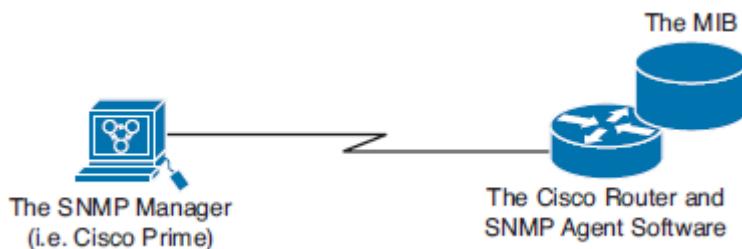
- מוגדרות כתובות IP וירטואליות שונות.
- מספר הקבוצה שמוגדר בנתבים שונה.
- בנתבים פועלת גרסה HSRP שונה.

Simple Network Management Protocol (SNMP)

הפרוטוקול שיר לשכבות היחסום והוא מאפשר להתקנים לשתף מידע על המצב הנוכחי שליהם במטרה לאטר בעיות. רוב הכלים לאייסוף ועיבוד מידע משתמשים ב프וטוקול זה. בנוסף, הפרוטוקול מאפשר לשנות חלק מההגדרות של ההתקן.

רכיבי SNMP

- SNMP מכיל שלושה רכיבים:
 - NMS (Network Management System) - נקרא גם SNMP Manager זו התוכנה שדרוכה ניתן לתקשר עם ההתקנים באמצעות SNMP ולקבל מידע על ההתקנים.
 - SNMP Agent - זה השירות שמותקנת בהתקן ומאפשר לו SNMP Manager לתקשר עם ההתקן.
 - MIB (Management Information Base) - זהו בסיס נתונים שמכיל אוסף של אובייקטים (OID) שונים לניהול ולמעקב. OID מייצג ערך מסוים לדוגמה: עומס על המעבד, טמפרטורת המעבד ועוד. חלק מהערכים ניתן לשנות. SNMP Agent מנהל את בסיס נתונים. SNMP Manager מבקש מהסוכן מידע והסוכן מביא לו את המידע מבוסיס הנתונים.



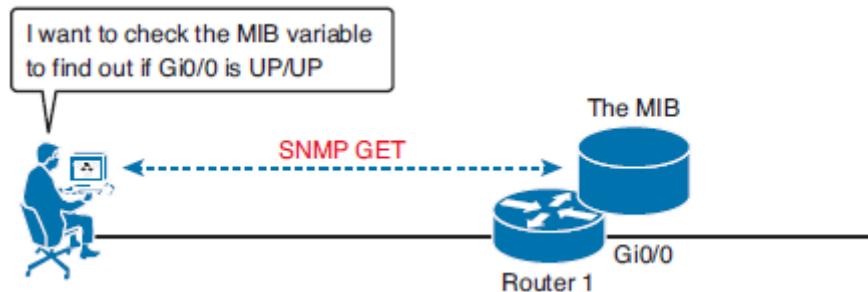
גרסאות SNMP

- SNMP יש שלוש גרסאות. הנה סיכום של שלושת הגרסאות:
 - SNMPv1 - יצא בשנת 1988 ונדריך לראות אותה מיושמת היום. הפרוטוקול תומך באימות ללא הצפנה ומשתמש רק ב- UDP.
 - SNMPv2c - יצא בשנת 1993 והציג כמה שיפורים שימושיים אך לא בתחום האבטחה. הפרוטוקול תומך באימות ללא הצפנה (community strings).
 - SNMPv3 - יצא בשנת 1998, השינוי הרלוונטי ביותר זה תמייה באימות מוצפן באמצעות MD5 או שלמות הנתונים באמצעות DES או DES-256. בנוסף משתמש ב- TCP.

פניות של SNMP

אחד הסיבות להצלחת הפרטוקול היא פשוטות הפקודות שבתוכם הוא תומך. קיימות מעט פקודות, אבל הם גמישות מספיק כדי לענות על הדרישות.

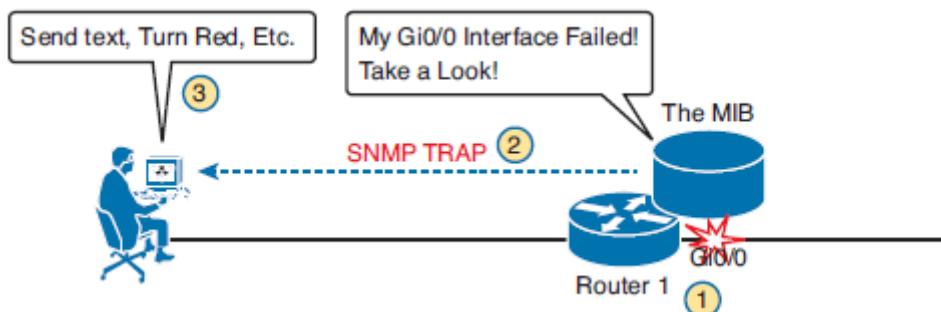
.Manager - בקשה של ערך מסוים (OID). הבקשת נשלחת ל- Agent על ידי



.GetNext - בקשה של האובייקט הבא ב- MIB. זהו י דרך לקבלת מידע מבסיס נתונים (MIB), בצורה מהירה.

.GetBulk - בקשה שמאפשרת לקבל מידע רב במהירות, כמו בקשות GetNext מרובות.

.Trap - מלכודת מוגדרת מראש בהתקן ומטרתה להודיע ל- Manager על אירוע חריג.



.INFORM - כמו Trap אך כולל acknowledgment.

.Set - זו פעולה הכתיבה היחידה שהוגדרה על ידי הפרטוקול. הודעה שנשלחת על ידי Manager ל- Agent כדי לשנות ערך של OID מסוים.

Monitoring applications

PRTG זו תוכנה מומלצת מאוד.

קיימת גרסה חינמית שכוללת שימוש בעד 30 חישנים.

:SNMP Ports

- port udp 161 - בקשה מידע מהתקן
- traps port udp 162 - לצורך



הגדלה SNMP

הגדרת הכתובת של SNMP Manager
ללא ניתמך ב-.packet tracer

Router(config)#snmp-server host {1.2.3.4}

הגדרת סיסמה ומצב (RO/RW)
SNMP יודע לעבוד בשני מצבים:

- Read-only (RO) - מאפשר גישת קרייה בלבד לאובייקטים ב-MIB.
- Read-write (RW) - מאפשר גישת קרייה/כתיבה לאובייקטים ב-MIB.

ניתן לחת שני סיסמות, אחת לקרייה (RO) והשנייה לכתיבה (RW).

Router(config)#snmp-server community {password} {RO/RW} {ACL}

ניתן להגביל את הגישה המותרת באמצעות ACL.

דוגמא:

```
R1(config)# ip access-list standard ACL_PROTECTSNMP
R1(config-std-nacl)# permit host 10.10.10.101
R1(config-std-nacl)# exit
R1(config)# snmp-server community V0olleyB@11111 RO ACL_PROTECTSNMP
```



Virtual Private Network

חיבור בין רשותות מרוחקות באמצעות האינטרנט יוצר שני בעיות:

- משאבי הרשת המרוחקת לא זמינים. לא ניתן לפנות לכתובות פרטיות מרוחקות.
- האינטרנט הוא מקום ציבורי ולכן מידע שנשלח דרכו חשוף לציטוט (Sniffing).

VPN יודע לתת מענה לשני הבעיה:

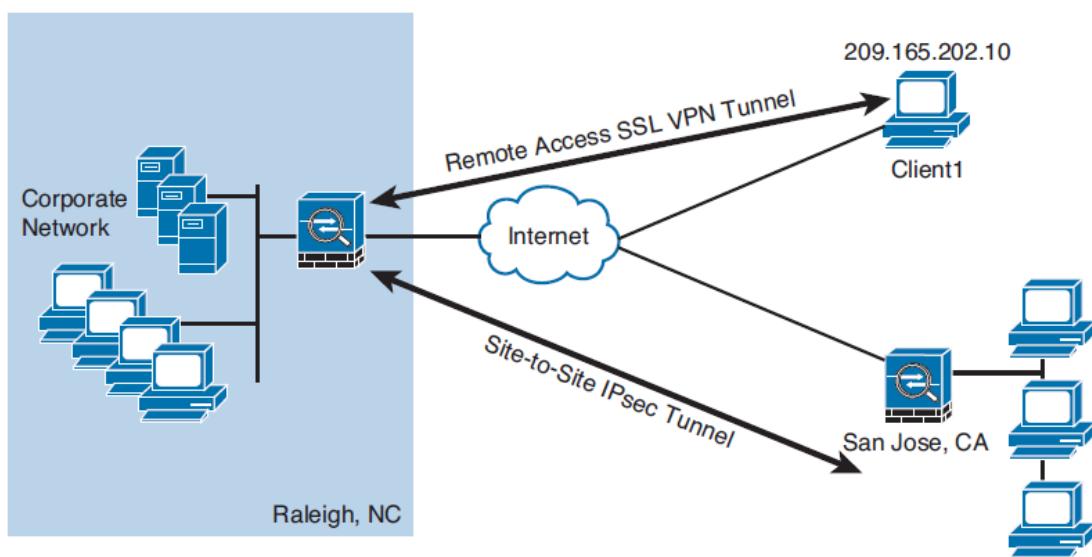
- VPN לחבר את המחשב לרשות המרוחקת כאילו הוא מחובר ישירות לרשת LAN.
- VPN מצפין את המידע וכך, רק מי שモරשה יכול לפעنه את המידע.

יתרונות VPN

- **אבטחה** - המידע שמועבר באמצעות VPN מאובטח על-ידי IPSec או SSL.
- **חיסכון כספי** - השימוש ב- VPN מונע את העלויות הגבוהה של קווים מושכרים (WAN) וכן דרך חסכנות לחיבור בין סניפים או בין עובד מרוחק למקום העבודה.
- **יכולת גידול** - בקלות ו מהירות ניתן לחבר סניפים ומשתמשים מרוחקים נוספים.

קיימים שני סוגי עיקריים של VPN:

- **Remote Access VPN** - מאפשר לך לחבר למרוחק לתוך הארגון. כך להלן יכול לגשת בצוירה מאובטחת למשאים בתוך הארגון.
- **Site-to-Site VPN** - חיבור בין סניפים. יצירת קשר ישיר ומאובטח בין שני סניפי החברה. ההצפנה מתבצעת רק ביציאת ה-packets לאינטרנט.



VPN Protocols

שלושת ה프וטוקולים (tunneling protocols) בעזרתם ניתן ליצור קשר מאובטח המ:

Point-to-Point Tunneling Protocol (PPTP)

- זהו פרוטוקול קצר מישן ופחות מאובטח מהפרוטוקולים האחרים.
יודע להצפין PPP packets. אחד מהחולשות של הפרוטוקול זה שההילך הקמת הקשר לא מוצפן. רק לאחר הקמת הקשר הפרוטוקול מוצפן את המידע.
Microsoft Point-to-Point Encrypting (MPPE) משתמש בפרוטוקול PPTP להצפנת המידע. עובד ב-TCP port 1723.

Layer 2 Tunneling Protocol (L2TP)

- יודע להצפין PPP packets. ברוב המקרים יעשה שימוש בפרוטוקול זה.
L2TP משתמש בפרוטוקול IPsec Internet Protocol Security (IPSec) להצפנת המידע. עובד ב-UDP port 1701.

Secure Socket Tunneling Protocol (SSTP)

- SSTP משתמש בפרוטוקול SSL Secure Sockets Layer (SSL) להצפנת המידע.
השימושHTTPS באפשר לעבר Firewalls בצורה פשוטה הרבה יותר מאשר שימוש ב-PPTP או L2TP. SSTP ניתן החל מ-Windows Server 2008 ו-Windows Vista SP1. השירות חייב תעודת דיגיטלי.

שלושת הפרוטוקולים משתמשים ב-PPP Point-to-Point Protocol (PPP) לביצוע user authentication.

VPN Authentication Protocols

כאשר מיישמים VPN, יש לבחור אחד מפרוטוקולים אלה לצורכי אימות שם משתמש וסיסמה של ה-Client. תהליך ההזדהות תחיליה מנסה להשתמש בפרוטוקול המאובטח ביותר המאפשר אצל ה-Client.

Password Authentication Protocol (PAP)

- לא מוצפן את הסיסמא (plaintext) ולכן עדיף לא להשתמש בו.

Shiva Password Authentication Protocol (SPAP)

- משתמש בהצפנה חלשה מסוג reversible encryption (לא נחשב כהצפנה טובה).
הפרוטוקול לא מוצפן את DATA.

Challenge Handshake Authentication Protocol (CHAP)

- משתמש בא�ורייתם הצפנה מסוג MD5 Message Digest 5 (MD5). CHAP לא מוצפן את DATA. DATA רק נחתם דיגיטלי (hash) ולכן פרוטוקול זה לא נחשב כפרוטוקול הצפנה.

- Microsoft CHAP (**MS-CHAP**) •
משתמש בהצפנה מסווג MPPE ומאפשר להצפין את ה- DATA. חזק ההצפנה
כמורכבות הסיסמה.
- MS-CHAP version 2** •
רמת אבטחה גבוהה יותר מ- MS-CHAP (תוקנו בו בעיות אבטחה).
חזק ההצפנה לא קשור למורכבות הסיסמה.
- Extensible Authentication Protocol-Transport Level Security •
USB - smart cards - certificates (EAP-TLS)
.Key

Internet Protocol Security (IPSec)

IPSec או חבילת פרוטוקולים שפועלת בשכבה שלישית של מודל OSI ונפוץ מאוד בשימוש ב-VPN. IPSec מבטח תעבורת רשת מסוג IP/TCP וכך להגן על המידע שזורם ברשת.

כיצד IPSec מבטח את זרימת הנתונים ברשת?

זהוי (Authentication)

זהוי הדרי של שני הצדדים לפני וטור כדי שידור הנתונים.

ניתן לבצע אימות במספר דרכים שונות:

- Pre-Shared secret key - שימוש בסיסמה לצורך זהוי.
- Public Key Infrastructure - בשיטה זו ההזדהות מתבצעת באמצעות תעודת דיגיטלית וזוג מפתחות (Public and private key).
- Remote Access VPN – משתמש ב-User authentication

סודיות (Confidentiality)

רק הצדדים שמתקשרים דרך VPN, יכולים להבין את הנתונים שנשלחו. אם מישחו מוצותה למשתמש, הוא יוכל לראות את החבילות אבל זה חסר משמעות כי התוכן של החבילה לא מובן (cipher text) כי אין לו את יכולת פענחו את הנתונים.

כך נראה תוכן של מידע מוצפן

```
Tp uijt jt uif tfdsfu nfttbhf. Ju jt fbtz up ef-fodszqu jg zpv lopx uif lfz.
```

האלגוריתמים והנוסחאות להצפנת נתונים זמינים בפורמבית וידועות לכלם. החלק שגורם להודיעו להיות סודית זה המפתח המשמש להצפנת הנתונים.

```
"So this is the secret message. It is easy to de-encrypt if you know the key."
```

שלמות המידע (Data Integrity)

חתימת המידע (Data signing) לא מאפשרת את זיהוף המידע.

כלומר, אם יבוצע שינוי במידע, זה יתגלה.

(Anti-Replay) packet

כל חבילה ממוסרת באמצעות Sequence Number. בדיקת ה-Sequence Number לא מאפשרת לתוקף ללכוד חבילה שמודרנה ולנסות לשדר אותה שוב במטרה לפתח session או לקבל גישה למשאים ברשת.

מחסנית הפרוטוקולים של IPSec

מחסנית הפרוטוקולים IPSec, מורכבת מפרוטוקולים רבים שמתעדכנים עם הזמן. כך IPSec יהיה רלוונטי גם בעתיד כי המחסנית שלו ניתנת לעדכון ורמת אבטחת המידע יכולה להשתנות בהתאם לצרכים של המשתמש.

Negotiation Protocol

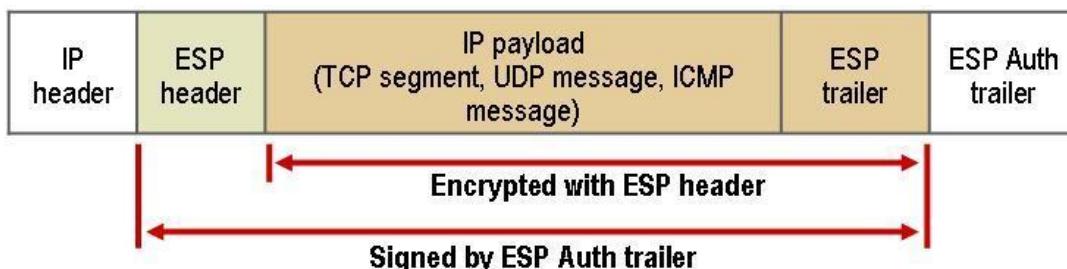
ניתן להשתמש בכל אחד מהפרוטוקולים בנפרד או עם שנייהם ביחד.

Authentication Header (AH) protocol

פרוטוקול זה מספק:

- זיהוי
- שלמות המידע
- חסימת שידור חוזר של packet
- לא תומך ב NAT

הפרוטוקול לא מבצע הצפנה ולכן המידע ניתן לקרוא אף לא ניתן לשינוי.

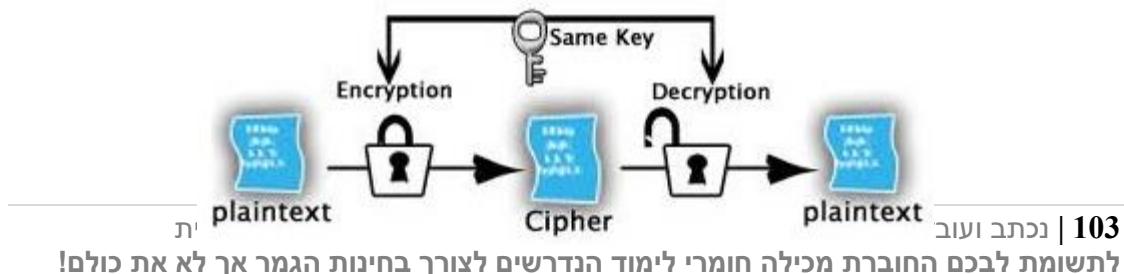


שיטות הצפנה

קיימות שני שיטות הצפנה:

Symmetric

בשיטת זו, קיימן מפתח אחד והוא משמש להצפנה ולפיענוח. שני הצדדים משתמשים באותו מפתח (Shared Secret). הצפנה סימטרית מהירה יותר מהצפנה אסימטרית ולא פחות חזקה ממנה.



הצפנה סימטריות משתמשת באחת משתי השיטות הבאות:

- block cipher - מצפין בכל פעם מקטע נתונים אחר.
- stream cipher - מצפין מידע זורם, byte אחר byte.

אלגוריתמים שימושיים ב-Symmetric key:

משמשים בעיקר להצפנה מידע מאוחסן (סטטי).

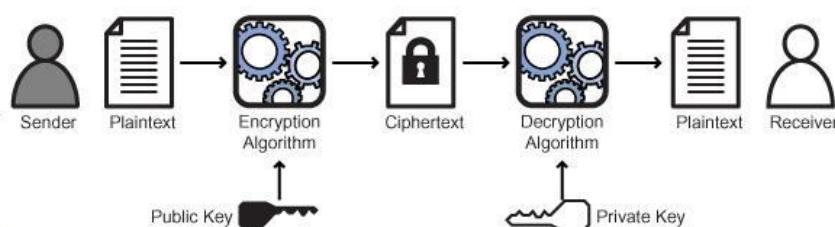
- DES (Data Encryption Standard) - נוצר על ידי IBM (מפתח באורך bits 56). בשימוש החל כמעט שנות השבעים. כיום הוא חשב ללא מאובטח בגל המפתח הקצר שלו.
- 3DES (Triple DES) - נוצר על ידי IBM (מפתח באורך bits 168 3×56). יוצר שלושה מפתחות לכל בלוק נתונים ומצפין את הבלוק שלושה פעמים (כל פעם עם מפתח אחר).
- AES (Advanced Encryption Standard) - מפתח באורך 128/192/256. ממלשת ארצות הברית מסогת את סוג ההצפנה של AES-256 כ- סוד' ביוטר.

Asymmetric

-

בשיטת זו, ההצפנה והפענוח מתרבעים באמצעות מפתחות שונים. מפתח ציבורי (Public-key) משמש להצפנה ומפתח פרטי (Private-key) משמש לפענוח. המפתח הציבורי מופץ לצורה חופשית לכל מי שanon רצים שישלחו לו מידע מוצפן. כאשר המידע מגיע למחשב שלנו, המידע מופיע במאזות המפתח הציבורי. בקרה זו המידע מוגן כי רק מי שיש לו את המפתח פרטי יכול לפענוח את המידע.

Public Key Encryption



הצפנה ציירית Asymmetric מאד חזקה אך מעמיסה מאוד על המעבד ולכן IPsec משלב בהצפנה את שני השיטות. בשלב הראשון השולח מצפין את המידע באמצעות הצפנה סימטרית. בשלב השני השולח מצפין את מפתח ההצפנה הסימטרי תוך שימוש בהצפנה אסימטרית. חשוב לציין – לכל קוביץ נוצר מפתח סימטרי חדש.

אלגוריתמים שימושיים ב-Asymmetric key:

משמשים בעיקר להצפנה מידע זורם או להצפנה תהליך היזדהות.

- RSA – מפתח באורך 512/1024 או יותר. משמש בעיקר ב- SSH.
המפתח מסוג Asymmetric-key
- DH (Diffie–Hellman) – מפתח באורך 512/1024 או יותר. משמש בעיקר ב- VPN.
המפתח מסוג Asymmetric-key

Data Integrity Protocol (Hashing)

- MD5 – מפתח באורך 128 bits
- SHA-1 – מפתח באורך 160 bits

Sוגי Authentication

קביעת סוג Authentication תלוי בטופולוגיה המחשבים וברמת האבטחה הנדרשת.

Kerberos V5

פרוטוקול בירית המחדל שמשמש ל Authentication בסביבת Active Directory trusted domains ניתן לשימוש בין מחשבים שנמצאים באותו trusted domains. שימוש ב프וטוקול זה לא דורש הגדרות נוספות כאשר מתבצעת תקשורת בין מחשבים שמוצרים על Domain.

Public Key Infrastructure

בשיטת זו ההזדהות מתבצעת באמצעות תעודה דיגיטלית. השתמש בשיטה זו במידע וביצונו לקיום תקשורת בין מחשבים שלא נמצאים באותו trusted domains (דרי' האינטראנט, שותפים עסקיים וכו'). כדי לישם שיטה זו יש צורך בפחות trusted certification authority אחד.

Pre-Shared secret key

שימוש בסיסמא לצורכי זהה. שיטה זו מספקת הci פחות אבטחה ובנוסף הסיסמא מאוחסנת בצורה לא מוצפנת במחשב או ב Active Directory. יש להשתמש בשיטה זו רק בסביבת ניסוי.

השלבים בתקשורת בין שני מחשבים המשתמשים ב- IPSec:

1. על המחשבים מיושמת מדיניות מקומית (local policy) או מדיניות זר GPO. המדיניות מגדרה באמצעות Filter על איזה סוג של תובנות רשות המדיניות תחול, ובעזרת Filter Actions נקבע באיזו צורה תוגן תובנות הרשות.
2. ה프וטוקול IKE (Internet Key Exchange) מפעיל באיזה שיטה להשתמש במקרה שהמחשבים יזהו אחד את השני. (certificate, Kerberos, preshared key).
3. מידע מוצפן או נתחת לפני שיודרו ברשת.

הגדמות ב- firewall

בשימוש בפרוטוקולים הבאים, יש לפתח את ה- ports.

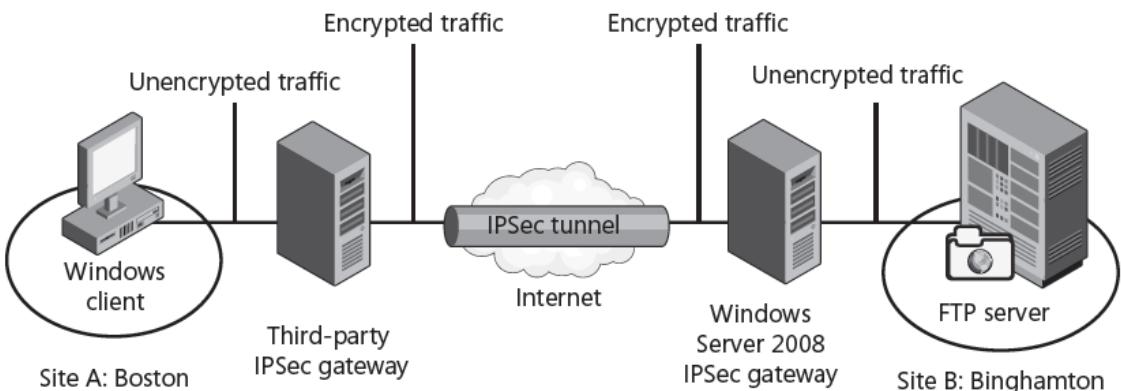
- TCP 51 – AH
- TCP 50 - ESP
- UDP 500 – IKE

Transport Mode

כברית מחדל IPsec עובד ב mode transport, כלומר אבטחה מקצתה רקaza. שיטה זו משמשת מחשבים בתחום LAN כדי לאבטחה את התקשרותם.

Tunnel Mode

ב Tunnel Mode האבטחה היא לא עד תחנת הקצה. בדרך כלל נשתמש ב- site to site vpn במקומ בשיטה זו.



IPSec Security Policy

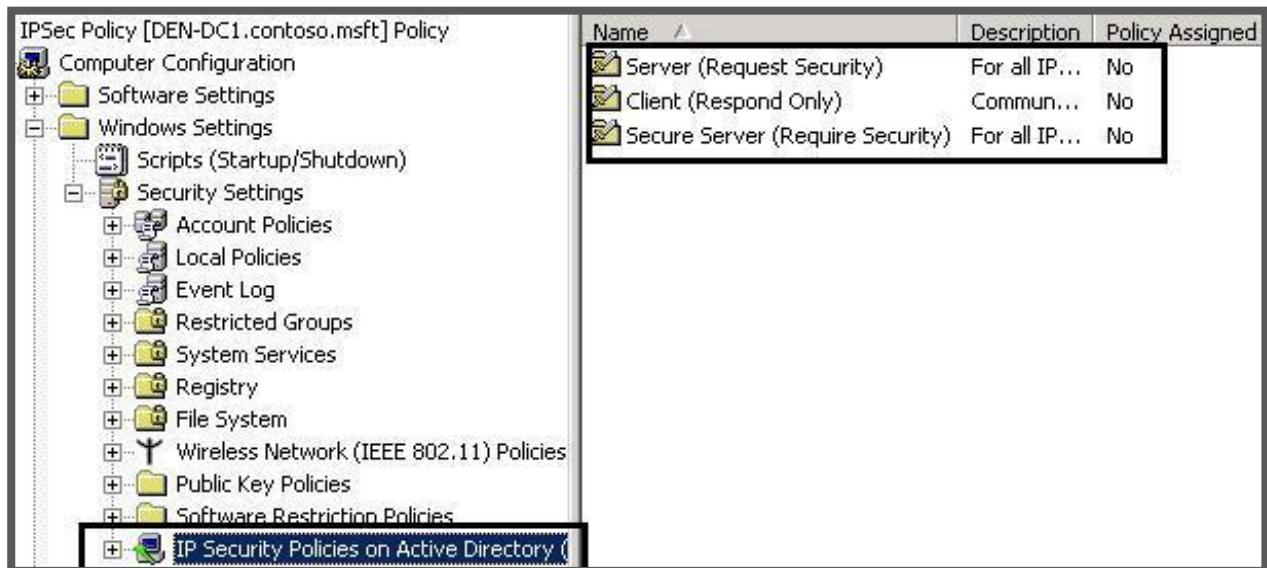
זו סידרת חוקים שקובעים לאיזה סוג של תעבורת רשות להתייחס (filters) ואיזה סוג של הגנה לישים (block, permit, negotiate security). ניתן לישם רק מדיניות אחת למחשב ולכון כל החוקים צריים להיות באותה מדיניות. לא ניתן לקבוע את סדר החוקים. החוקים היוצרים ספציפיים הראשונים ולאחר מכן החוקים הכללים יותר.

IPsec Policy

логічн:		
	IP Filter Lists	Filter Actions
Less specific/Lower priority	Policy Rule #1 Filter #1: Telnet Traffic from 192.168.3.32 Filter #2: POP3 Traffic from 192.168.3.200	Negotiate Security (Require Encryption)
	Policy Rule #2 Filter #1: All Telnet Traffic Filter #2: All POP3 Traffic	Block
	Policy Rule #3 Filter #1: All Traffic	Negotiate Security (Request Authentication)

(מדיניות ברירת המחדל) Default IPSec Policies

- Client (Respond Only) – שימוש ב- IPSec ב- Client (Respond Only) רק אם מחשב היעד דורש זאת.
- בדרך כלל נשתמש ב Policy זה על Clients.
- Server (Request Security) – המחשב ידרש שימוש ב- IPSec אך יתקשר גם אם אין למחשב היעד אפשרות להשתמש ב- IPSec.
- Secure Server (Require Security) – המחשב ידרש שימוש ב- IPSec ולא ייצור קשר עם מחשב שלא מסוגל להשתמש ב- IPSec. כל תעבורת הרשות היוצאת תהיה מוצפנת.



יצירת חוק בעזרת Security Rule Wizard

פרוט חקת המשת שלבים ביצירת חוק חדש:

1. Tunnel endpoint – אם נעשה שימוש ב- Tunneling, יש צורך לציין לעבורה יצאת את הכתובת IP של השרת הקרוב ליעד שמבצע את ה- Tunneling.

2. Connection type – הגדרה זו משמשת להגבלת סוג החיבור (LAN, Remote).

access (allow) החוק מתייחס.

3. IP Filter list – הגדרה זו קובעת לאיזה סוג תעבורת רשות החוק תהיה.

כברירת מחדל קיימים שני סוגי Filters All IP Traffic ו All ICMP Traffic.

על-ידי לחיצה על הכפתור Add, ניתן ליצור Filters נוספים.

4. Filter action – איזה פעולה יש לבצע על תעבורת הרשות.

Permit (Request Security), Optional (Request Security), Request (Request Security).

5. Authentication methods – באיזה שיטה יש להשתמש לשורף Authentication methods.

Kerberos, preshared key (certificate from a specified certification authority).

